FC

10

AD

BRL CR 302

BRL

CONTRACT REPORT NO. 302

DESCRIPTION OF FORTRAN PROGRAM DAWNA FOR ANALYSIS OF MUZZLE BLAST FIELD

AD A024485

Prepared by

Advanced Technology Laboratories, Inc.
Merrick & Stewart Avenues
Westbury, New York 11590

April 1976

Approved for public release; distribution unlimited.

USA BALLISTIC RESEARCH LABORATORIES
ABERDEEN PROVING GROUND, MARYLAND

Destroy this report when it is no longer needed.
Do not return it to the originator.

Secondary distribution of this report by originating or sponsoring activity is prohibited.

Additional copies of this report may be obtained from the National Technical Information Service, U.S. Department of Commerce, Springfield, Virginia 22151.

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

18

1. REPORT NUMBER 19 CR-
BRL CONTRACT REPORT ~~NO.~~ 302

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle) 6
Description of FORTRAN Program DAWNA for Analysis of Muzzle Blast Field.

5. TYPE OF REPORT & PERIOD COVERED
9 Contract rept.

6. PERFORMING ORG. REPORT NUMBER 14
ATL-TM-184

7. AUTHOR(s) 10
J. Ranlet ~~and~~ J. Erdos

8. CONTRACT OR GRANT NUMBER(s) 15
DAAD05-74-C-0784 NEW

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Advanced Technology Labo~~ratorie~~s, Inc.
Merrick & Stewart Aves
Westbury, New Y~~ork~~ 11590

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
12 75p.

11. CONTROLLING OFFICE NAME AND ADDRESS
USA Ballistic Research Laboratories
Aberdeen Proving Ground, Maryland 21005

12. REPORT DATE 11
APR~~IL 1~~976

13. NUMBER OF PAGES
76

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
US Army Materiel Development & Readiness Command
5001 Eisenhower Avenue
Alexandria, VA 22333

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Muzzle Blast Field
Transitional Ballistics
Acoustic Waves
Spherical Blast Wave Theory 4
Unsteady Aerodynamics

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The FORTRAN IV Program DAWNA described in this report solves the set of partial differential equations governing the flow on the axis of symmetry between the blast wave and the Mach disc of a muzzle blast field. A complete description of the method of solution of the governing equations and statement of the boundary conditicns can be found in BRL Contractor's Report No. 297.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED 404 568

TABLE OF CONTENTS

		Page
I.	INTRODUCTION	1
II.	PROGRAM STRUCTURE	2
III.	DESCRIPTION OF INPUT	8
	A. PUNCHED CARD FORMAT	8
	B. RESTART CAPABILITY	13
IV.	DESCRIPTION OF OUTPUT	15
	A. OUTPUT FORMAT	15
	B. IDENTIFICATION OF VARIABLES	16
	C. SAMPLE INPUT AND OUTPUT	16
V.	PROGRAM LISTING	22
	DISTRIBUTION LIST	71

LIST OF FIGURES

		Page
FIG. 1.	SCHEMATIC OF GRID NETWORK	10
FIG. 2.	CDC SCOPE 2.1 OPERATING SYSTEM CONTROL CARDS ILLUSTRATING THE USE OF THE PROGRAM RESTART CAPABILITY	14
FIG. 3.	TYPICAL INPUT IN CARD IMAGE FORMAT	17
FIG. 4.	FIRST PAGE OF PROGRAM OUTPUT	18
FIG. 5.	SECOND PAGE OF PROGRAM OUTPUT	19
FIG. 6.	TYPICAL OUTPUT OF FINITE DIFFERENCE CALCULATION	20
FIG. 7.	TYPICAL OUTPUT OF ACOUSTIC WAVE ANALYSIS	21

SECTION I

INTRODUCTION

The FØRTRAN IV program DAWNA solves the set of partial differential equations governing the flow on the axis of symmetry between the blast wave and the Mach disc of a muzzle blast field. A complete description of the method of solution of the governing equations and statement of the boundary conditions is given in Reference (1). Program DAWNA is a revised edition of program DAWN previously described in Reference (2). The two main refinements included in program DAWNA are the following:

1) An initialization technique has been developed that allows the computer program to self-start without the use of empirical relationships (such as the initial locations of the flow discontinuities).

2) An acoustic analysis which enables the continuation of the blast field calculation to very late times.

Besides the major analytical refinements presented above, various other modifications have been included in the present computer code which minimize possible sources of numerical error as well as making the program more convenient for the user. The more pertinent of these include the use of automatic grid subdivision, a streamline trace for the determination of the properties upstream of the Mach disc and the option of choosing the units of both the input and output parameters.

SECTION II
PROGRAM STRUCTURE

The main program acts an as executive routine which calls the principal subroutines, as follows. Subroutine INDATA is called once to define the finite difference grid and initialize the dependent variables. Subroutines SHØCK, CØNTACT and INT PT are called sequentially to advance the solution from time t to time t+Δt. The dependent variables are then reinitialized, printed and the above sequence repeated. This loop is terminated when the selected number of time steps have been completed, at which time the final solution can be written out in TAPE12 (at the option of the user) and the program execution is stopped.

Subroutines SHØCK and CØNTACT accomplish the solution at the two shocks and the contact surface by the method of characteristic technique described in Reference (1). The three surfaces of discontinuity are identified by the indices 1, 2 and 3 which refer to the Mach disc, contact surface and blast wave, respectively. Subroutine INT PT accomplishes the solution at the interior grid points by the finite difference algorithm devised by MacCormack (Reference 3). Within subroutine INT PT the index K=1 denotes the solution between the Mach disc and contact and K=2 denotes the solution between the contact and the blast wave. The indices LØØP=0 and 1 refer to the first and second iterates of the MacCormack algorithm.

The subroutines associated with the acoustic wave computation are called from subroutine ACØUS which only returns control to the main program at the completion of the run.

The functions of all the subroutines are summarized in Table I. The main program variables are identified in Table II.

TABLE I

SUBROUTINES AND FUNCTIONS

NAME	DESCRIPTION
1. CL	Calculates flow properties on the centerline of the supersonic jet exhaust plume from the muzzle exit conditions and the centerline Mach number at the location of the Mach disc.
2. CØNTACT	Calculates the position and velocity of the contact surface and the flow properties on both sides of the contact surface.
3. FS	Sets the flow variables on the upstream side of the blast wave to the specified ambient (i.e., "infinity") conditions.
4. INDATA	Reads the input data and defines accordingly the finite difference grid and the initial values of the dependent variables.
5. INT PT	Computes the new finite difference solution at the interior grid points and the new location of the grid points at each succeeding time step.
6. LPØINT	Locates the intersection of a characteristic surface and a time plane, and performs the necessary interpolations of variables from the solution at the grid points.
7. MUZZLE	Determines the pressure, speed of sound and Mach number at the muzzle exit as a function of time from a spline-fit of the corresponding tabular input data.

NAME	DESCRIPTION
8. PUNCH	Writes the final time step on TAPE12 for restarting the calculation (see Section III(B) for an explanation of the restart capability of the program).
9. SETN	Reinitializes the arrays in which the new solution at time t+Δt will be stored.
10. SHØCK	Calculates the position and velocity of a moving shock and the flow properties on the downstream side of the shock. The index K in the calling sequence is used to denote the Mach disc (K=1) and the blast wave (K=3).
11. SPLINE	Performs a spline-fit of tabular data (see Reference 4).
12. SPLINT	Uses the spline-fit coefficients to interpolate data at arbitrary values of the independent argument. (First and second derivatives of the dependent variable are also calculated, but not used in the present program.)
13. STEP	Computes the maximum allowable time step, Δt, based on the Courant, Friedrichs and Lewy criterion.
14. INTER	Interpolates data when adding or deleting a grid point.
15. INIT	Determines the initial location and fluid properties of the Mach disc, contact surface and blast wave.
16. ACØUS	Calls the various subroutines associated with the integration of the acoustic wave equation.
17. INT1A	Performs the integration of the acoustic wave equation using a fourth-order Runke-Kutta method.

NAME		DESCRIPTION
18.	DERSUB	Subroutine which evaluates the derivative used in the Runge-Kutta integration.
19.	CHSUB	Dummy subroutine called from subroutine INT1A.

TABLE II

PRINCIPAL PROGRAM VARIABLES

FØRTRAN NAME	DEFINITION
IR(1)	Number of grid points in region 1 (i.e., from Mach disc to the contact).
IR(2)	Total number of grid points (including points on both sides of both shocks and the contact except for the free stream side of the blast wave)
Z	Axial distance (in feet).
P	Natural logarithm of pressure (in psf).
U	Gas velocity (in fps).
S	Entropy (in ft^2/sec^2-^oR)
A	Speed of sound (in fps).
RH	Gas density (in $slugs/ft^3$).
W	Velocity of surface of discontinuity.
GAM1, GAM2	Ratio of specific heats for regions 1 and 2, respectively.

FØRTRAN NAME	DESCRIPTION
CP1, CP2	Specific heat at constant pressure for regions 1 and 2, respectively.
RJET	Radius of gun bore.
XME	Muzzle exit Mach number.
PE	Muzzle exit pressure (in psf).
AE	Sound speed at muzzle exit (in fps).
PINF	Ambient pressure (in psf).
UINF	Ambient gas velocity (in fps) (usually 0.0).
AINF	Speed of sound in ambient gas (in fps).
RHINF	Ambient gas density (slugs/ft^3).
TIME	Current time (seconds).
TIMEF	Final time (seconds).
KK	Maximum number of time steps.
KO	Current number of completed time steps.
LL	Interval in the number of time steps between printing of complete flow field solution.
IPUNCH	Index for option to write final solution on TAPE12.

FØRTRAN NAME	DESCRIPTION
DT	Time increment.
DZ	Increment in axial distance.

Definition of other program variables should be self-evident from the context of their usage.

SECTION III
DESCRIPTION OF INPUT

A. PUNCHED CARD FORMAT

Card Number	Format	Columns	Description
1	I5	1-5	starting step of run
	I5	6-10	final step of run
	I5	11-15	print interval
	I5	16-20	output file creation index (0 - no output file created; 1 - output file created on TAPE12)
	I5	21-25	restart indicator (0 - initial program submittal; 1 - program reads initial step from TAPE10)
	I5	26-30	option for moving origin of coordinate system (0 - fixed region; 1 - moving origin)
	I5	31-40	input unit index - specifies the units of the input (0 - metric; 1 - english)
	I5	41-45	output unit index - determines the units of the output (0 - metric; 1 - english; 2 - non-dimensional)
	I5	46-50	acoustic wave index (0 - normal program operation; 1 - only acoustic wave calculation performed)
2	E10.0	1-10	starting time of run (seconds)
	E10.0	11-20	final time of run (seconds)
	E10.0	21-30	multiplicative factor on maximum time step calculated from stability theory (recommended value is 1.0)

Card Number	Format	Columns	Description
2	E10.0	31-40	multiplicative factor on minimum grid spacing as determined by initial number of grid points (recommended value is $\leq$ 1.0)
	E10.0	41-50	multiplicative factor on minimum grid spacing to determine maximum grid spacing (recommended value is $\geq$ 2.0)
3	E10.0	1-10	radius of gun bore (m or ft)
	E10.0	11-20	coordinate system index (0 - planar; 1 - cylindrical; 2 - spherical)
	E10.0	21-30	ratio of distance from muzzle to origin of coordinate system to the distance from muzzle to the Mach disc

If the restart indicator is equal to one on card number 1, card number 4 is not required.

Card Number	Format	Columns	Description
4	I5	1-5	last data point in region 1 (shown as IR1 in Figure 1)
	I5	6-10	last data point in region 2 (shown as IR2 in Figure 1)
	E10.0	11-20	specific heat ratio in region 1
	E10.0	21-30	Specific heat ratio in region 2
	E10.0	31-40	specific heat at constant pressure in region 1 (m^2/sec^2 ^oK or ft^2/sec^2 ^oR)

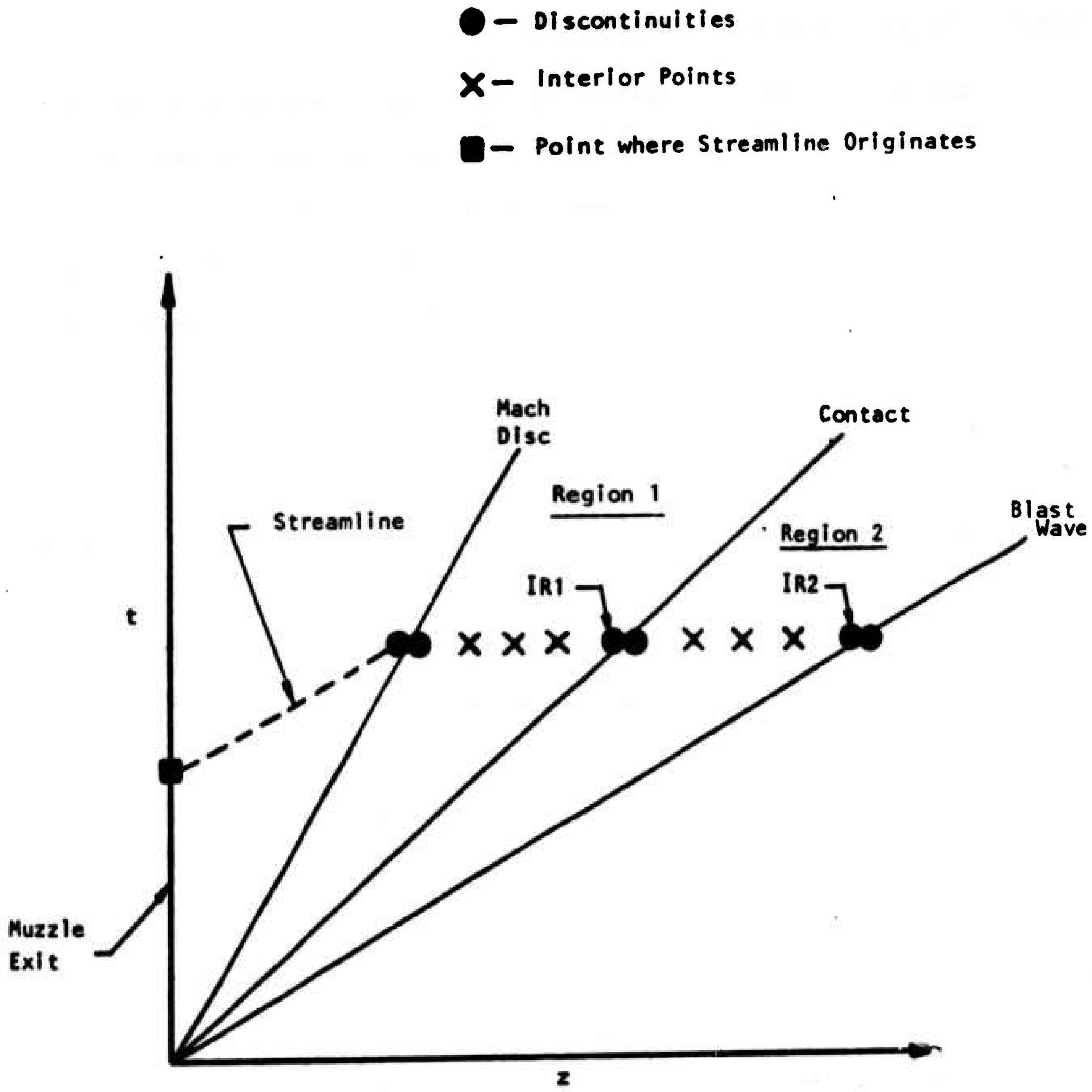

FIGURE 1. SCHEMATIC OF GRID NETWORK

Card Number	Format	Columns	Description
4	E10.0	41-50	specific heat at constant pressure in region 2 ($m^2/sec^2\ {}^oK$ or $ft^2/sec^2\ {}^oR$)
5	E10.0	1-10	ambient pressure (N/m^2 or lb/ft^2)
	E10.0	11-20	ambient gas velocity (m/sec or ft/sec) (typically 0.0)
	E10.0	21-30	ambient speed of sound (m/sec or ft/sec)
	E10.0	31-40	blast wave Mach number at which acoustic wave analysis commences
6	I5	1-5	INUM-number of points in table of muzzle exit properties (maximum of 25 points)
7	There are "INUM" of the following cards describing the muzzle exit properties.		
7a	E10.0	1-10	time (seconds)
	E10.0	11-20	pressure (N/m^2 or lb/ft^2)
	E10.0	21-30	speed of sound (m/sec or ft/sec)
	E10.0	31-40	Mach number
8	I5	1-5	INUMP-number of points in table describing the plume centerline Mach number distribution

ard umber	Format	Columns	Description
9	The following cards describe the plume centerline Mach number distribution. A maximum of 35 points are allowed. Input the location of the point (ZZ) followed by the centerline Mach number (ZMCL) for that point. Input 4 pairs to a card and continue on the next card until all "INUMP" points have been input. A sample of the first card is shown below:		
9a	E10.0	1-10	ZZ(1) - axial distance from muzzle in bore radii
	E10.0	11-20	ZMCL(1) - plume centerline Mach number at distance ZZ(1) from nozzle
	E10.0	21-30	ZZ(2)
	E10.0	31-40	ZMCL(2)
	E10.0	41-50	ZZ(3)
	E10.0	51-60	ZMCL(3)
	E10.0	61-70	ZZ(4)
	E10.0	71-80	ZMCL(4)

If the acoustic wave index is equal to zero on card 1, card number 10 is not required.

10	E10.0	1-10	the location of the contact surface (m or ft)
	E10.0	11-20	the location of the blast wave (m or ft)
	E10.0	21-30	the initial time step to be used by the Runge-Kutta integration routine (usually taken to be the final time step obtained by the stability requirements of the finite difference portion of the program) (seconds)

Card Number	Format	Columns	Description
10	E10.0	31-40	the constant in the Mach disc equation printed out at the last finite difference step
	E10.0	41-50	the constant in the blast wave equation printed out at the last finite difference step
	E10.0	51-60	the constant in the velocity curve fit printed out at the last finite difference step

B. Restart Capability - The program possesses the capability of using a solution previously stored on disk or tape to initialize the necessary variables at each grid point. To create such a file, the output file creation index (card 1) must be set equal to one (1). When the final time step is reached the necessary variables are output on TAPE12 and may be saved by the appropriate use of control cards.

To make use of this file, the restart indicator (card 1) should be set equal to one (1). With the exception of card 2 which must be changed to correspond to the time step saved on the file and card 4 which must be omitted, the program input deck is the same. The variables are then read from TAPE10 and are used for the initiation of the program. At the completion of this run, the variables can again be output on TAPE12 if desired. Care must be taken to assure that the files and tapes correspond to the desired input and output data. Typical control cards that illustrate the use of the restart capability are shown in Figure (2).

CREATION OF OUTPUT FILE

```
JOB CARD
CHARGE CARD
REQUEST(TAPE12,*PF)
ATTACH(LGO,DAWN,CY=1,ID=ATLXXX)
MAP(PART)
LGO.
REWIND(TAPE12)
CATALOG(TAPE12,NAMESTEPXXX,CY=1,ID=ATLXXX)
EOR
```

RESTART USING OUTPUT FILE

```
JOB CARD
CHARGE CARD
REQUEST(TAPE12,*PF)
ATTACH(LGO,DAWN,CY=1,ID=ATLXXX)
MAP(PART)
ATTACH(TAPE10,NAMESTEPXXX,CY=1,ID=ATLXXX)
REWIND(TAPE10)
LGO.
REWIND(TAPE12)
CATALOG(TAPE12,NAMESTEPXXXX,CY=1,ID=ATLXXX)
EOR
```

FIGURE 2. CDC SCOPE 2.1 OPERATING SYSTEM CONTROL CARDS ILLUSTRATING THE USE OF THE PROGRAM RESTART CAPABILITY

SECTION IV
DESCRIPTION OF OUTPUT

A. Output Format - The first page of the program output prints a narrative which informs the user of various input parameters such as the initial and final time steps, print interval, free stream properties etc. On the second page of output, the input values of the muzzle exit conditions and the plume centerline Mach number distribution are printed out in tabular form. The units of the input variables are specified by the input unit index as described previously in Section III.

On the following pages of the printout, starting with the initial step, the fluid dynamic properties are printed at each grid point. As a convenience to the user the locations of the Mach disc, contact surface and blast wave are specified in the output by the letters MD, C and BW respectively. At intermediate time steps where a complete output is not desired the velocity and position of the discontinuities are printed. The program has the capability of printing out the fluid variables in either metric or English units or in a non-dimensional form. The output unit index described in Section III controls which unit option is used. A table illustrating the unit option is contained in the following sub-section.

The form of the output for the acoustic wave computation is similar to that of the finite difference portion of the program except that only non-dimensional output is used. When the program reaches the time for switching to the acoustic analysis (i.e., when $W(3)/a_\infty \leq 1.10$), the final finite difference time step is output along with the necessary parameters for resubmitting an acoustic wave run.

B. Identification of Variables

STEP - total number of steps taken

TIME - total elapsed time (seconds)

	Description	Output Unit Index 0	1	2
Z	axial distance	meter	feet	Z/D
P	pressure	N/m^2	lb/ft^2	P/P_∞
U	velocity	m/sec	ft/sec	U/a_∞
S	entropy	$m^2/sec^2\ ^oK$	$ft^2/sec^2\ ^oR$	$(S-S_{ref})/C_v$
A	speed of sound	m/sec	ft/sec	a/a_∞
RH	density	kg/m^3	$slug/ft^3$	ρ/ρ_∞
TEMP	temperature	^oK	^oR	T/T_∞
MACH	Mach number	-	-	-
W(1)	velocity of Mach disc	m/sec	ft/sec	$W(1)/a_\infty$
W(2)	velocity of contact	m/sec	ft/sec	$W(2)/a_\infty$
W(3)	velocity of blast wave	m/sec	ft/sec	$W(3)/a_\infty$
Z(1)	location of Mach disc	meter	feet	Z(1)/D
Z(2)	location of contact	meter	feet	Z(2)/D
Z(3)	location of blast wave	meter	feet	Z(3)/D

C. Sample Input and Output

```
       0    10     2     0     0     0     1     2     0
0.0         0.3   E-021.0          1.0         2.0
0.00913     2.0         0.40
      7   131.25        1.40       9507.08     6006.0
2120.       0.0         1120.      1.10
   21
.876   E-031.2981E+064.2493E+031.
.924   E-031.1776E+063.9386E+031.
.972   E-031.0918E+063.7152E+031.
1.02   E-031.0172E+063.5246E+031.
1.068  E-039.5173E+053.3547E+031.
1.117  E-038.9409E+053.2008E+031.
1.165  E-038.4269E+053.0591E+031.
1.213  E-037.9668E+052.9272E+031.
1.261  E-037.4033E+052.8504E+031.
1.309  E-036.7934E+052.8140E+031.
1.357  E-036.2439E+052.7841E+031.
1.405  E-035.7510E+052.7592E+031.
1.453  E-035.3093E+052.7379E+031.
1.502  E-034.9135E+052.7192E+031.
1.550  E-034.5585E+052.7031E+031.
1.598  E-034.2389E+052.6890E+031.
1.646  E-033.9509E+052.6765E+031.
1.694  E-033.6913E+052.6650E+031.
1.742  E-033.4561E+052.6552E+031.
2.876  E-038.9808E+042.3228E+031.
3.876  E-033.2268E+042.0932E+031.
   25
0.0         1.0         0.2048     1.0371      0.4072      1.1541      0.6164      1.3203
1.0240      1.6874      1.454      2.0468      1.8198      2.3315      2.254       2.6268
2.916       2.9952      3.478      3.2655      4.256       3.5857      4.958       3.834
5.788       4.1182      6.45       4.2777      7.736       4.6329      8.854       4.8369
10.0        5.15        11.0       5.26        12.0        5.42        14.0        5.72
16.0        6.02        18.0       6.3         20.0        6.58        40.0        8.5
68.0        10.0
```

FIGURE 3. TYPICAL INPUT IN CARD IMAGE FORMAT

M U Z Z L E   B L A S T   A N A L Y S I S

```
STARTING STEP =   0
FINAL STEP =  10
PRINT INTERVAL =  10
STARTING TIME =  0.            SEC.
FINAL TIME =   .30000E-02  SEC.
LAST PT. REGION NO. 1 =   7
LAST PT. REGION NO. 2 =  13
MOVING COORDINATE SYSTEM OPTION =    0
RADIUS OF THE JET =   .91300E-02  FT.
FLOW INDEX = 2
ORIGIN OF SPHERICAL SYSTEM =  .400E+00 TIMES RJET
FREE STREAM PRESSURE =   .21200E+04  LBS/FT/FT
FREE STREAM VELOCITY =  0.          FT/SEC
FREE STREAM SPEED OF SOUND =   .11200E+04  FT/SEC
SPECIFIC HEAT RATIO (REGION 1) =1.25
SPECIFIC HEAT RATIO (REGION 2) =1.40
SPECIFIC HEAT AT CONSTANT PRESSURE(REGION 1) = 9507.1  FT**2/SEC**2-DEG.R
SPECIFIC HEAT AT CONSTANT PRESSURE(REGION 2) = 6006.0  FT**2/SEC**2-DEG.R
BARREL EXIT PRESSURE =   .12981E+07  LBS/FT/FT
BARREL EXIT MACH NO. =   .10000E+01
BARREL EXIT SPEED OF SOUND =   .42493E+04  FT/SEC
```

FIGURE 4. FIRST PAGE OF PROGRAM OUTPUT

MUZZLE EXIT CONDITIONS

TIME	PRESSURE	SOUND SP	MACH
0.	.1298E+07	.4249E+04	.1000E+01
.4800E-04	.1178E+07	.3939E+04	.1000E+01
.9600E-04	.1092E+07	.3715E+04	.1000E+01
.1440E-03	.1017E+07	.3525E+04	.1000E+01
.1920E-03	.9517E+06	.3355E+04	.1000E+01
.2410E-03	.8941E+06	.3201E+04	.1000E+01
.2890E-03	.8427E+06	.3059E+04	.1000E+01
.3370E-03	.7967E+06	.2927E+04	.1000E+01
.3850E-03	.7403E+06	.2850E+04	.1000E+01
.4330E-03	.6793E+06	.2814E+04	.1000E+01
.4810E-03	.6244E+06	.2784E+04	.1000E+01
.5290E-03	.5751E+06	.2759E+04	.1000E+01
.5770E-03	.5309E+06	.2738E+04	.1000E+01
.6260E-03	.4914E+06	.2719E+04	.1000E+01
.6740E-03	.4559E+06	.2703E+04	.1000E+01
.7220E-03	.4239E+06	.2689E+04	.1000E+01
.7700E-03	.3951E+06	.2677E+04	.1000E+01
.8180E-03	.3691E+06	.2665E+04	.1000E+01
.8660E-03	.3456E+06	.2655E+04	.1000E+01
.2000E-02	.8981E+05	.2323E+04	.1000E+01
.3000E-02	.3227E+05	.2093E+04	.1000E+01

PLUME CENTERLINE MACH NUMBER

X/R	MACH
0.	.1000E+01
.2048E+00	.1037E+01
.4072E+00	.1154E+01
.6164E+00	.1320E+01
.1024E+01	.1687E+01
.1454E+01	.2047E+01
.1820E+01	.2332E+01
.2254E+01	.2627E+01
.2916E+01	.2995E+01
.3478E+01	.3266E+01
.4256E+01	.3586E+01
.4958E+01	.3834E+01
.5788E+01	.4118E+01
.6450E+01	.4278E+01
.7736E+01	.4633E+01
.8854E+01	.4837E+01
.1000E+02	.5150E+01
.1100E+02	.5260E+01
.1200E+02	.5420E+01
.1400E+02	.5720E+01
.1600E+02	.6020E+01
.1800E+02	.6300E+01
.2000E+02	.6580E+01
.4000E+02	.8500E+01
.6800E+02	.1000E+02

FIGURE 5. SECOND PAGE OF PROGRAM OUTPUT

STEP = 0 TIME = .74650E-05 SEC.

I		Z	P	U	S	A	RH	TEMP	MACH
1	MD	.28578E-01	.45046E+05	.94081E+04	0.	.30331E+04	.61206E-02	.38706E+04	.31018E+01
2	MD	.28578E-01	.55556E+05	.88934E+04	.21043E+01	.30977E+04	.72371E-02	.40373E+04	.28709E+01
3		.30268E-01	.55556E+05	.81068E+04	.21043E+01	.30977E+04	.72371E-02	.40373E+04	.26170E+01
4		.31958E-01	.55556E+05	.73202E+04	.21043E+01	.30977E+04	.72371E-02	.40373E+04	.23631E+01
5		.33649E-01	.55556E+05	.65336E+04	.21043E+01	.30977E+04	.72371E-02	.40373E+04	.21092E+01
6		.35339E-01	.55556E+05	.57470E+04	.21043E+01	.30977E+04	.72371E-02	.40373E+04	.18553E+01
7	C	.37030E-01	.55556E+05	.49605E+04	.21043E+01	.30977E+04	.72371E-02	.40373E+04	.16013E+01
8	C	.37030E-01	.55556E+05	.49605E+04	.44499E+04	.25865E+04	.11625E-01	.27850E+04	.19178E+01
9		.37574E-01	.55556E+05	.48166E+04	.44499E+04	.25866E+04	.11625E-01	.27850E+04	.18621E+01
10		.38118E-01	.55556E+05	.46728E+04	.44499E+04	.25866E+04	.11625E-01	.27850E+04	.18065E+01
11		.38662E-01	.55556E+05	.45289E+04	.44499E+04	.25866E+04	.11625E-01	.27850E+04	.17509E+01
12		.39207E-01	.55556E+05	.43851E+04	.44499E+04	.25866E+04	.11625E-01	.27850E+04	.16953E+01
13	BW	.39751E-01	.55556E+05	.42412E+04	.44499E+04	.25866E+04	.11625E-01	.27850E+04	.16397E+01
14	BW	.39751E-01	.21200E+04	0.	0.	.11200E+04	.23661E-02	.52214E+03	0.

W(1)	W(2)	W(3)	Z(1)	Z(2)	Z(3)	TIME	DT
.60717E+04	.49605E+04	.53250E+04	.28578E-01	.37030E-01	.39751E-01	.74650E-05	0.
.60854E+04	.49732E+04	.53192E+04	.28863E-01	.37263E-01	.40001E-01	.75120E-05	.46991E-07
.61163E+04	.49857E+04	.53134E+04	.29149E-01	.37496E-01	.40250E-01	.75587E-05	.46791E-07
.61465E+04	.49979E+04	.53079E+04	.29433E-01	.37728E-01	.40496E-01	.76052E-05	.46437E-07
.61760E+04	.50097E+04	.53025E+04	.29717E-01	.37958E-01	.40741E-01	.76513E-05	.46084E-07
.62049E+04	.50212E+04	.52973E+04	.30000E-01	.38188E-01	.40983E-01	.76970E-05	.45731E-07
.62331E+04	.50324E+04	.52922E+04	.30283E-01	.38416E-01	.41224E-01	.77424E-05	.45380E-07
.62614E+04	.50433E+04	.52874E+04	.30565E-01	.38643E-01	.41462E-01	.77874E-05	.45029E-07
.62891E+04	.50538E+04	.52826E+04	.30847E-01	.38868E-01	.41698E-01	.78321E-05	.44668E-07
.63151E+04	.50641E+04	.52781E+04	.31126E-01	.39092E-01	.41932E-01	.78764E-05	.44308E-07

FIGURE 6. TYPICAL OUTPUT OF FINITE DIFFERENCE CALCULATION

STEP =1095 TIME = .30000E-02

	Z/D	P	RHO	U
MD	.39490E+01			
	.18401E+03	.10000E+01	.10000E+01	0.
	.18568E+03	.10006E+01	.10011E+01	.20501E-02
	.18738E+03	.10016E+01	.10024E+01	.42083E-02
	.18912E+03	.10032E+01	.10040E+01	.64911E-02
	.19088E+03	.10053E+01	.10061E+01	.89150E-02
	.19268E+03	.10081E+01	.10085E+01	.11496E-01
	.19452E+03	.10117E+01	.10113E+01	.14252E-01
	.19638E+03	.10159E+01	.10146E+01	.17204E-01
	.19829E+03	.10207E+01	.10182E+01	.20381E-01
	.20023E+03	.10260E+01	.10220E+01	.23811E-01
	.20221E+03	.10316E+01	.10258E+01	.27524E-01
	.20423E+03	.10374E+01	.10296E+01	.31547E-01
	.20630E+03	.10432E+01	.10333E+01	.35899E-01
	.20840E+03	.10492E+01	.10369E+01	.40539E-01
	.21055E+03	.10553E+01	.10405E+01	.45406E-01
	.21274E+03	.10617E+01	.10442E+01	.50438E-01
	.21497E+03	.10684E+01	.10482E+01	.55575E-01
	.21726E+03	.10754E+01	.10525E+01	.60755E-01
	.21959E+03	.10830E+01	.10572E+01	.65970E-01
BW	.22198E+03	.10910E+01	.10623E+01	.71290E-01

END CONDITION MET

FIGURE 7. TYPICAL OUTPUT OF ACOUSTIC WAVE ANALYSIS

SECTION V
PROGRAM LISTING

```
      PROGRAM DAWNA(INPUT,OUTPUT,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,
     1TAPE10,TAPE12)
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
      COMMON/G/KK,LL,KO,TIME,TIMEF
      COMMON/IPU/ IPUNCH
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      COMMON/GA/GAM1,GAM2,CP1,CP2
      COMMON/BLAST/IBW,BWCON,SPEED,RJET2
      COMMON/DEBUG/IDEBUG
      COMMON/UNITS/IUNIT,IUDUM,IUNOUT,IUOUT,FTME,PUNIT,DEGRK,RHUNIT
      COMMON/BWANAL/BWMACH
      DATA H1/2H  /,H2/2HMD/,H3/2H C/,H4/2HBW/
      DT=0.0
      CALL INDATA
      FACINF=CP2*(GAM2-1.0)/GAM2
      FACT2=FACINF
      FACT1=CP1*(GAM1-1.0)/GAM1
    3 CONTINUE
      IR1=IR(1)
      IR1P=IR1+1
      IR2=IR(2)
      IR2P=IR2+1
      IF(KO.GE.KK) GO TO 4
      IF((KO/LL)*LL.NE.KO) GO TO 5
    4 CONTINUE
      WRITE(6,7) KO,TIME
    7 FORMAT(1H1,10X,*STEP =*,I4,10X,*TIME =*,E13.5,1X,4HSEC.,//)
      WRITE(6,8)
    8 FORMAT(4X*I*11X,*Z*,13X,*P*,13X,*U*,13X,*S*,13X,*A*,12X,*RH*,
     111X,*TEMP*,10X,*MACH*)
      I1=IR(2)+1
      FDUM=AINF*TIME
      DO 9 I=1,I1
      FACREG=FACT1
      IF(I.GT.IR1) FACREG=FACT2
      H5=H1
      IF(I.EQ.1.OR.I.EQ.2) H5=H2
      IF(I.EQ.IR1.OR.I.EQ.IR1P) H5=H3
      IF(I.GE.IR2) H5=H4
      GO TO(22,24,26),IUOUT
   22 CONTINUE
      Z5=Z(I)*FTME
      P5=EXP(P(I))*PUNIT
      U5=U(I)*FTME
      A5=A(I)*FTME
      R5=RH(I)*RHUNIT
      S5=S(I)*FTME**2*DEGRK
      TEMP=(P5/(R5*FACREG))/FTME**2/DEGRK
      GO TO 50
   24 CONTINUE
      Z5=Z(I)
      P5=EXP(P(I))
      U5=U(I)
      A5=A(I)
```

```
      R5=RH(I)
      S5=S(I)
      TEMP=(P5/(R5*FACREG))
      GO TO 50
   26 CONTINUE
      Z5=Z(I)/RJET2
      P5=EXP(P(I))/PINF
      U5=U(I)/AINF
      A5=A(I)/AINF
      R5=RH(I)/RHINF
      TEMP=(P5/R5)*FACINF/FACREG
      IF(I.GT.IR1) GO TO 30
      S5=GAM1*(S(I)-S(1))/CP1
      GO TO 40
   30 CONTINUE
      S5=GAM2*(S(I)-S(IR2P))/CP2
   40 CONTINUE
   50 CONTINUE
      XM5=U(I)/A(I)
      WRITE(6,10) I,H5,Z5,P5,U5,S5,A5,R5,TEMP,XM5
 2000 FORMAT(8X,5E14.5)
    9 CONTINUE
   10 FORMAT(I5,1XA2,8E14.5)
      WRITE(6,12)
   12 FORMAT(/)
      IF(KO.GE.KK) GO TO 6
      WRITE(6,21)
   21 FORMAT(11X*W(1)*10X*W(2)*10X*W(3)*10X*Z(1)*10X*Z(2)*10X*Z(3)*
     110X*TIME*,11X,*DT*)
    5 CONTINUE
      Z1=Z(1)/RJET2
      Z2=Z(IR1)/RJET2
      Z3=Z(IR2)/RJET2
      IF(Z3.GT.1.0)
     1BWCON=Z3     *(1.0-(AINF/W(3))**2)*SQRT(ALOG(Z3      ))
      IF(Z3.GT.1.0) BWTCON=TIME*1.0E+06+(RJET2/AINF)*(BWCON*
     1SQRT(ALOG(Z3))-Z3)*1.0E+06
      ZMD=0.69*XME*SQRT(GAM1*PE/EXP(P(2)))
      GO TO(32,34,36),IUOUT
   32 CONTINUE
      W1DUM=W(1)*FTME
      W2DUM=W(2)*FTME
      W3DUM=W(3)*FTME
      Z1DUM=Z(1)*FTME
      Z2DUM=Z(IR1)*FTME
      Z3DUM=Z(IR2)*FTME
      GO TO 38
   34 CONTINUE
      W1DUM=W(1)
      W2DUM=W(2)
      W3DUM=W(3)
      Z1DUM=Z(1)
      Z2DUM=Z(IR1)
      Z3DUM=Z(IR2)
      GO TO 38
   36 CONTINUE
      W1DUM=W(1)/AINF
```

```
      W2DUM=W(2)/AINF
      W3DUM=W(3)/AINF
      Z1DUM=Z(1)/RJET2
      Z2DUM=Z(IR1)/RJET2
      Z3DUM=Z(IR2)/RJET2
   38 CONTINUE
      WRITE(6,11) W1DUM,W2DUM,W3DUM,Z1DUM,Z2DUM,Z3DUM,TIME,DT
   11 FORMAT(5X,8E14.5,2F8.2)
      IF(IBW.EQ.0) GO TO 14
      LL=LLDUM
      ZCON=Z(IR1)
      ZBW=Z(IR2)
      RCT=Z(IR1P)-FDUM
      SGN=SIGN(1.0,RCT)
      DO 60 I=IR1P,IR2
      RCT=Z(I)-FDUM
      SGN1=SIGN(1.0,RCT)
      IF(SGN.EQ.SGN1) GO TO 59
      IRCT=I
      IRCTM=I-1
      RCTM=Z(IRCTM)-FDUM
      TERM=(Z(IRCT)/RJET2)**2*(RH(IRCT)/RHINF-1.0-U(IRCT)/SPEED)
      TERMM=(Z(IRCTM)/RJET2)**2*(RH(IRCTM)/RHINF-1.0-U(IRCTM)/SPEED)
      RATRCT=RCT/(RCT-RCTM)
      RHORCT=TERM+RATRCT*(TERMM-TERM)
      GO TO 70
   59 CONTINUE
      SGN=SGN1
   60 CONTINUE
      RHORCT=1.0
   70 CONTINUE
      WRITE(6,1000) CONMD,BWCON,RHORCT
 1000 FORMAT(//,1X,*THE CONSTANT IN THE MACH DISC EQUATION IS*,E13.5,
     1//,1X,*THE CONSTANT IN THE BLAST WAVE EQUATION IS*,E13.5,
     1//,1X,*THE CONSTANT IN THE VELOCITY CURVE FIT IS*,E13.5)
      CALL ACOUS(ZCON,ZBW,DT,CONMD,RHORCT)
   14 CONTINUE
      CALL SETN
      CALL STEP
      CALL SHOCK(1)
      CALL SHOCK(3)
      CALL CONTCT(2)
      CALL INT PT
      IF(IMOVE.EQ.0) GO TO 20
       ZZERO=ZN(1)*FACT
       DZZERO=WN(1)*FACT
       DTZZER=(WN(1)-W(1))/DT
   20 CONTINUE
      I1=IR(2)+1
      DO 1 I=1,I1
      Z (I)=Z N(I)
      P (I)=P N(I)
      U (I)=U N(I)
      S (I)=S N(I)
      A (I)=A N(I)
    1 RH(I)=RHN(I)
      DO 2 I=1,3
```

```
    2 W(I)=WN(I)
      KO=KO+1
      TIME=TIME+DT
      IF((A(3)/AINF).LE.BWMACH) IBW=1
      IF(IBW.EQ.0) GO TO 3
      CALL MUZZLE(TIME,PEXIT,AEXIT,XMEXIT)
      CONMD=Z(1)/(XMEXIT*SQRT(GAM1*PEXIT/PINF))/RJET2
      LLDUM=LL
      LL=1
      GO TO 3
    6 CONTINUE
      IF(IPUNCH.NE.0) CALL PUNCH
      STOP
      END
```

```
      SUBROUTINE CL(IT1,BT)
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/G/KK,LL,KO,TIME,TIMEF
      COMMON/MCL/ZZ(35),ZMCL(35),EMCL(35),INUMP
      COMMON/INIT/PM(25),AM(25),XMM(25),TM(25)
      COMMON/DEBUG/IDEBUG
      DIMENSION ZM(1),XM(1),DXM(1),D2XM(1)
      DATA MAX,ERRVAL/1,0.001/
      ZE=0.0
      ITER=0
      RATIO=0.5
      IF(BT.EQ.0.0) UN(1)=U(1)
      TINIT=TIME+DT
      T1=TM(1)
      IF(IDEBUG.EQ.6) WRITE(6,1) IT1,ZE,RATIO,TIME,DT,TINIT,T1,BT
   20 CONTINUE
      TDUMS=RATIO*TINIT+(1.0-RATIO)*T1
      CALL MUZZLE(TDUMS,PE,AE,XME)
      UE=0.5*(XME*AE+UN(1))
      TDUMP=TINIT-(ZN(1)-ZE)/UE
      ERR=(TDUMS-TDUMP)/TDUMS
      IF(IDEBUG.EQ.6) WRITE(6,1) ITER,TDUMS,TDUMP,PE,AE,XME,UE,ZN(1),ERR
      IF(TDUMP.LT.T1.OR.TDUMP.GT.TINIT) GO TO 100
      IF(ABS(ERR).LT.ERRVAL) GO TO 300
      ITER=ITER+1
      IF(ITER.GT.20) GO TO 200
      RATIO=(T1-TDUMP)/(T1-TINIT)
      GO TO 20
  100 CONTINUE
      WRITE(6,1000)
 1000 FORMAT(1X,*STREAMLINE IN SUBROUTINE CL IS OUT OF BOUNDS*)
      CALL EXIT
  200 CONTINUE
      WRITE(6,2000)
 2000 FORMAT(1X,*TOO MANY ITERATIONS IN SUBROUTINE CL*)
      CALL EXIT
  300 CONTINUE
      GAM=GAM1
      I=1
      ZZ1=ZN(I)/RJET+1.
      G1=GAM-1.
      XME2=XME*XME
      F1=1.+G1*XME2/2.
      G3=GAM/G1
      G4=1./G1
      ZM(1)=ZZ1
      CALL SPLINT(ZZ,ZMCL,INUMP,ZM,MAX,XM,DXM,D2XM,EMCL)
      XMCL=XM(1)
      XMCL2=XMCL*XMCL
      F2=1.+G1*XMCL2/2.
      PPE=(F1/F2)**G3
      RHRE=(F1/F2)**G4
      PN(I)=PPE*PE
```

```
      RHE=GAM*PE/AE/AE
      RHN(I)=RHRE*RHE
      AN(I)=SQRT(GAM*PN(I)/RHN(I))
      UN(I)=XMCL*AN(I)
      SN(I)=S(1)+CP1*((ALOG(PN(I))-P(1))/GAM1-ALOG(RHN(I)/RH(1)))
      PN(I)=ALOG(PN(I))
      IF(IDEBUG.EQ.6) WRITE(6,1) I,PN(I),RHN(I),AN(I),UN(I),SN(I),
     1S(1),P(1),RH(1)
      RETURN
    1 FORMAT(1I5,9E13.5)
      END
```

```
      SUBROUTINE CONTCT(K)
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/C/ZI(200),PI(200),UI(200),RHI(200),SI(200),AI(200)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/E/BT,AL
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      I=IR(1)
      IT1=1
      AL=1.
      BT=0.
      WN(K)=W(K)
      UN(I)=WN(K)
      UN(I+1)=UN(I)
    2 ZN(I)=Z(I)+(AL* W(K)+BT*WN(K))*DT
    1 ZN(I+1)=ZN(I)
      U1=WN(K)
      U2=U1
      GAM=GAM1
      CP=CP1
      CALL LPOINT(I,1.,1,1)
      GAM=GAM2
      CP=CP2
      CALL LPOINT(I+1,-1.,-1,2)
      RN=ZN(I)-ZZERO
      R1=ZI(1)-ZZERO
      R2=ZI(2)-ZZERO
      A1=GAM1*(AL/AI(1)+BT/AN(I))
      B1=GAM2*(AL/AI(2)+BT/AN(I+1))
       IF(IMOVE.EQ.0) GO TO 200
      A4=GAM1*XJ*(AL*(UI(1)-DZZERO    )/R1+BT*(UN(I)-DZZERO     )/RN)
      A4=A4*DT
       B4=GAM2*XJ*(AL*(UI(2)-DZZERO    )/R2+BT*(UN(I+1)-DZZERO     )/RN)
      B4=B4*DT
      GO TO 210
  200 A4=(AL*UI(1)/R1+BT*UN(I)/RN)*DT*XJ*GAM1
      B4=(AL*UI(2)/R2+BT*UN(I+1)/RN)*XJ*GAM2*DT
  210 PC1=PI(1)-(U1-UI(1))*A1-A4
      PC2=PI(2)+(U2-UI(2))*B1-B4
      APC1=EXP(PC1)
      APC2=EXP(PC2)
      ER=(APC1-APC2)/(APC1+APC2)*2.0
      IF(ABS(ER).LT.1.E-04) GO TO 7
      IT1=IT1+1
      IF(IT1.LE.15) GO TO 777
      WRITE(6,111)
  111 FORMAT(5X,*TOO MANY ITERATIONS IN SUBROUTINE CONTCT*)
      STOP
  777 IF(IT1.GT.2) GO TO 14
      BET=WN(2)
      ER1=ER
      BET1=BET
      BET=1.01*BET
      GO TO 15
   14 DUM=BET1-ER1*(BET-BET1)/(ER-ER1)
      ER1=ER
```

```
      BET1=BET
      BET=DUM
   15 WN(K)=BET
      ZN(I)=Z(I)+.5*(W(K)+WN(K))*DT
      GO TO 1
    7 PN(I)=PC1
      PN(I+1)=PN(I)
      UN(I)=U1
      UN(I+1)=U2
      SN(I)=S(I)
      SN(I+1)=S(I+1)
      CK=ALOG(RH(I   ))-P(I   )/GAM1
      RHN(I   )=PN(I   )/GAM1-(SN(I   )-S(I   ))/CP1+CK
      RHN(I)=EXP(RHN(I))
      CK=ALOG(RH(I+1))-P(I+1)/GAM2
      RHN(I+1)=PN(I+1)/GAM2-(SN(I+1)-S(I+1))/CP2+CK
      RHN(I+1)=EXP(RHN(I+1))
      AN(I)=SQRT(GAM1*EXP(PN(I))/RHN(I))
      AN(I+1)=SQRT(GAM2*EXP(PN(I+1))/RHN(I+1))
      IF(BT.GT.0.) GO TO 9
      BT=.5
      AL=.5
      IT1=1
      GO TO 2
    9 RETURN
      END
```

```
SUBROUTINE FS
COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
COMMON/GA/ GAM1,GAM2,CP1,CP2
I=IR(2)+1
PN(I)=PINF
UN(I)=UINF
RHN(I)=RHINF
SN(I)=0.
AN(I)=SQRT(GAM2*PN(I)/RHN(I))
PN(I)=ALOG(PN(I))
RETURN
END
```

```
      SUBROUTINE INDATA
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
      COMMON/G/KK,LL,KO,TIME,TIMEF
      COMMON/DZ/DZMIN,DZMAX
      COMMON/DTSTAB/DTSTAB
      COMMON/IPU/ IPUNCH
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/FIT/EMP(25),EMA(25),EMM(25),INUM
      COMMON/INIT/PM(25),AM(25),XMM(25),TM(25)
      COMMON/MCL/ZZ(35),ZMCL(35),EMCL(35),INUMP
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      COMMON/PRAT/PRAT
      COMMON/DEBUG/IDEBUG
      COMMON/BLAST/IBW,BWCON,SPEED,RJET2
      COMMON/BWANAL/BWMACH
      COMMON/UNITS/IUNIT,IUDUM,IUNOUT,IUOUT,FTME,PUNIT,DEGRK,RHUNIT
      DIMENSION TD(1),PD(1),DPD(1),D2PD(1)
      DIMENSION H3(2),H4(2),H5(2),H6(2),H7(2)
      DATA H1/4HRJET/,H2/4HZ(1)/
      DATA IDEBUG/0/
      DATA FACP,FACA/1.0,1.0/
      DATA FTME,PUNIT,DEGRK,RHUNIT/0.3048,47.880258,1.8,515.379/
      DATA H8/4HSEC./
      DATA H3(1),H4(1),H5(1),H6(1),H7(1)/3HM, ,9HN/M**2    ,6HM/SEC ,
     110HM**2/SEC**,8H2-DEG.K /
      DATA H3(2),H4(2),H5(2),H6(2),H7(2)/3HFT.,9HLBS/FT/FT,6HFT/SEC
     1,10HFT**2/SEC*,8H*2-DEG.R/
C***********************************************************************
C     READ INPUTS
C***********************************************************************
      READ(5,1)     KO,KK,LL,IPUNCH,IRSTRT,IMOVE,IUNIT,IUNOUT,IBW
      IUDUM=IUNIT+1
      IUOUT=IUNOUT+1
      HDUM=H1
      IF(IMOVE.EQ.1) HDUM=H2
      READ(5,2) TIME,TIMEF,DTSTAB,DZMIN,DZMAX
      READ(5,2) RJET,XJ,FACT
      IF(IUNIT.EQ.0) RJET=RJET/FTME
      IF(IRSTRT.GT.0) GO TO 8
      READ(5,3) IR(1),IR(2),GAM1,GAM2,CP1,CP2
      IF(IUNIT.EQ.1) GO TO 8
      CP1=CP1/FTME**2/DEGRK
      CP2=CP2/FTME**2/DEGRK
    8 CONTINUE
      READ(5,2) PINF,UINF,AINF,BWMACH
      IF(IUNIT.EQ.1) GO TO 11
      PINF=PINF/PUNIT
      UINF=UINF/FTME
      AINF=AINF/FTME
      FACP=PUNIT
      FACA=FTME
   11 CONTINUE
      READ(5,1) INUM
      DO 10 I=1,INUM
```

```
      READ(5,2) TM(I),PM(I),AM(I),XMM(I)
      IF(I.EQ.1) TM1=TM(1)
      TM(I)=TM(I)-TM1
      IF(IUNIT.EQ.1) GO TO 10
      PM(I)=PM(I)/PUNIT
      AM(I)=AM(I)/FTME
   10 CONTINUE
      CALL SPLINE(TM,PM,INUM,EMP)
      CALL SPLINE(TM,AM,INUM,EMA)
      CALL SPLINE(TM,XMM,INUM,EMM)
      CALL MUZZLE(TIME,PE,AE,XME)
      READ(5,1) INUMP
      READ(5,2) (ZZ(I),ZMCL(I),I=1,INUMP)
      IF(IBW.EQ.0) GO TO 7
      READ(5,2) ZCON,ZBW,DT,CONMD,BWCON,RHORCT
      IF(IUNIT.EQ.1) GO TO 7
      ZBW=ZBW/FTME
      ZCON=ZCON/FTME
    7 CONTINUE
      IF(IRSTRT.GT.0) GO TO 300
    9 CONTINUE
      RHINF=GAM2*PINF/AINF/AINF
      SPEED=AINF
      RJET2=RJET*2.
      IF(IBW.EQ.1) CALL ACOUS(ZCON,ZBW,DT,CONMD,RHORCT)
      IF(IUNIT.EQ.1) GO TO 320
      CP1DUM=CP1*FTME**2*DEGRK
      CP2DUM=CP2*FTME**2*DEGRK
      PEDUM=PE*PUNIT
      AEDUM=AE*FTME
      RJDUM=RJET*FTME
      PIDUM=PINF*PUNIT
      UIDUM=UINF*FTME
      AIDUM=AINF*FTME
      GO TO 325
  320 CONTINUE
      CP1DUM=CP1
      CP2DUM=CP2
      PEDUM=PE
      AEDUM=AE
      RJDUM=RJET
      PIDUM=PINF
      UIDUM=UINF
      AIDUM=AINF
  325 CONTINUE
C***********************************************************************
C     WRITE INPUTS
C***********************************************************************
      WRITE(6,1632)
      WRITE(6,200) KO,KK,LL,TIME,H8,TIMEF,H8,IR,IMOVE
      J=XJ+.5
      WRITE(6,201) RJDUM,H3(IUDUM),J,FACT,HDUM
      WRITE(6,202) PIDUM,H4(IUDUM),UIDUM,H5(IUDUM),AIDUM,H5(IUDUM)
      WRITE(6,206) GAM1,GAM2,CP1DUM,H6(IUDUM),H7(IUDUM),CP2DUM,H6(IUDUM)
     1,H7(IUDUM)
      WRITE(6,207) PEDUM,H4(IUDUM),XME,AEDUM,H5(IUDUM)
      WRITE(6,205)
```

```
      WRITE(6,204)
      INUMQ=INUM
      IF(INUMP.LT.INUMQ) INUMQ=INUMP
      DO 326 I=1,INUMQ
      PDUMMY=PM(I)*FACP
      ADUMMY=AM(I)*FACA
      WRITE(6,500) TM(I),PDUMMY,ADUMMY,XMM(I),ZZ(I),ZMCL(I)
  326 CONTINUE
      INUMQ=INUMQ+1
      IF(INUM-INUMP) 307,308,309
  307 WRITE(6,501) (ZZ(I),ZMCL(I),I=INUMQ,INUMP)
      GO TO 308
  309 CONTINUE
      DO 327 I=INUMQ,INUM
      PDUMMY=PM(I)*FACP
      ADUMMY=AM(I)*FACA
      WRITE(6,502) TM(I),PDUMMY,ADUMMY,XMM(I)
  327 CONTINUE
  308 CONTINUE
      DO 12 I=1,INUMP
      ZZ(I)=ZZ(I)+1.0
   12 CONTINUE
C**********************************************************************
C     INITIALIZE FLOW PROPERTIES
C**********************************************************************
      CALL SPLINE(ZZ,ZMCL,INUMP,EMCL)
      IF(IRSTRT.GT.0) GO TO 342
      CALL INIT
      IR1=IR(1)
      IR11=IR1+1
      IR2=IR(2)
      IR21=IR(2)+1
      IR4=IR1-1
      IR5=IR1+2
      IR6=IR2-1
      P2=EXP(P(2))
      PIR1=EXP(P(IR1))
      CK=ALOG(RH(1))-P(1)/GAM1
      DO 330 I=3,IR4
      RAT =FLOAT(I-2)/FLOAT(IR1-2)
      Z(I)=Z(2)+(Z(IR1)-Z(2))*RAT
      U(I)=U(2)+(U(IR1)-U(2))*RAT
      S(I)=RAT*ALOG10(S(IR1)/S(2    ))
      S(I)=10.**S(I)
      S(I)=S(2    )*S(I)
      P(I)=P2+(PIR1-P2)*RAT
      RH(I)=ALOG(P(I))/GAM1-(S(I)-S(1))/CP1+CK
      RH(I)=EXP(RH(I))
      A(I)=SQRT(GAM1*P(I)/RH(I))
      P(I)=ALOG(P(I))
  330 CONTINUE
      P2=EXP(P(IR11))
      PIR2=EXP(P(IR2))
      CK=ALOG(RHINF)-ALOG(PINF)/GAM2
      DO 340 I=IR5 ,IR6
      RAT =FLOAT(I-IR11)/FLOAT(IR2-IR11)
      Z(I)=Z(IR11)+(Z(IR2)-Z(IR11))*RAT
```

```
      U(I)=U(IR11)+(U(IR2)-U(IR11))*RAT
      S(I)=RAT*ALOG10(S(IR2)/S(IR11))
      S(I)=10.**S(I)
      S(I)=S(IR11)*S(I)
      P(I)=P2+(PIR2-P2)*RAT
      RH(I)=ALOG(P(I))/GAM2-(S(I)-S(IR21))/CP2+CK
      RH(I)=EXP(RH(I))
      A(I)=SQRT(GAM2*P(I)/RH(I))
      P(I)=ALOG(P(I))
  340 CONTINUE
  342 CONTINUE
      RETURN
  300 CONTINUE
C*********************************************************************
C     READ RESTART VARIABLES
C*********************************************************************
      READ(10) IR(1),IR(2),GAM1,GAM2,CP1,CP2
      READ(10) (W(N),N=1,3)
      READ(10) ZZERO,DZZERO,DTZZER
      I1=IR(2)+1
      READ(10) (Z(I),P(I),U(I),RH(I),S(I),A(I),I=1,I1)
      IDUM1=IR(1)
      IDUM2=IR(2)
      ZDUM1=(Z(IDUM1)-Z(2))/FLOAT(IDUM1-2)
      ZDUM2=(Z(IDUM2)-Z(IDUM1))/FLOAT(IDUM2-IDUM1-1)
      DZMIN=DZMIN*AMIN1(ZDUM1,ZDUM2)*0.999
      DZMAX=DZMAX*DZMIN*1.001
      GO TO 9
    1 FORMAT(14I5)
    2 FORMAT(8E10.0)
    3 FORMAT(2I5,4E10.0)
   22 FORMAT(4E15.0)
  203 FORMAT(8E12.4)
 1632 FORMAT(1H1/////14X*M U Z Z L E  B L A S T  A N A L Y S I S*//)
  200 FORMAT(      9X,*STARTING STEP =*I4//9X,*FINAL STEP =*I4//9X*PRINT
     1INTERVAL =*I4//9X*STARTING TIME =*E13.5,2X,A4,
     2//9X*FINAL TIME =*E13.5,2X,A4,//9X
     3*LAST PT. REGION NO. 1 =*I4//9X,*LAST PT. REGION NO. 2 =*I4//
     4 9X*MOVING COORDINATE SYSTEM OPTION =*I5/)
  201 FORMAT(      9X*RADIUS OF THE JET =*E13.5,2X,A3,//9X
     1*FLOW INDEX =*I2//9X*ORIGIN OF SPHERICAL SYSTEM =*E10.3,* TIMES *
     1A4,/)
  202 FORMAT(   9X,*FREE STREAM PRESSURE =*E13.5,2X,A9,//9X,
     1*FREE STREAM VELOCITY =*E13.5,2X,A6,    //9X,*FREE STREAM SPEED OF
     1 SOUND =*E13.5,2X,A6,/)
  205 FORMAT(1H1///13X*MUZZLE EXIT CONDITIONS*19X*PLUME CENTERLINE MACH
     1NUMBER*/)
  204 FORMAT(5X*TIME*6X*PRESSURE*4X*SOUND SP*6X*MACH*16X*X/R*9X*MACH*)
  206 FORMAT(9X*SPECIFIC HEAT RATIO (REGION 1) =*F4.2,//9X*SPECIFIC HEAT
     1 RATIO (REGION 2) =*F4.2,//,9X,*SPECIFIC HEAT AT CONSTANT PRESSURE
     2(REGION 1) =*,F7.1,2X,A10,A8,
     3              //,9X*SPECIFIC HEAT AT CONSTANT PRESSURE(REGION
     42) =*,F7.1,2X,A10,A8,/)
  207 FORMAT( 9X,*BARREL EXIT PRESSURE =*E13.5,2X,A9,      //9X,*BARREL
     1 EXIT MACH NO. =*E13.5//9X*BARREL EXIT SPEED OF SOUND =*E13.5,2X
     2,A6)
  500 FORMAT(4E12.4,8X,2E12.4)
```

```
501 FORMAT(56X,2E12.4)
502 FORMAT(4E12.4)
    END
```

```
      SUBROUTINE INT PT
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/C/ZI(200),PI(200),UI(200),RHI(200),SI(200),AI(200)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/E/BT,AL
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      AL=.5
      BT=.5
      K=1
      GAM=GAM1
      CP=CP1
      NCL=2
      NCM=IR(1)
   88 CONTINUE
      NCL1=NCL+1
      NCM1=NCM-1
      M=NCM+1
      DO 20 I=NCL,NCM
      U I(I)=U (I)
      P I(I)=P (I)
      S I(I)=S (I)
      A I(I)=A (I)
   20 RHI(I)=RH(I)
      DX=1./FLOAT(NCM-NCL)
      LOOP=0
      DEL=ZN(NCM)-ZN(NCL)
      DD=1./DEL
    8 XI=0.
       IF(LOOP .EQ.1) GO TO 30
      WNK=W(K)
      WNKP=W(K+1)
      GO TO 31
   30 WNK=WN(K)
       WNKP=WN(K+1)
   31 CONTINUE
      DO 1 I=NCL1,NCM1
      XI=XI+DX
      NM1=I-LOOP
      NP1=NM1+1
      EE=DD*EXP(PI(I))/RHI(I)
      CC=DD*(UI(I)+(XI-1.)*WNK   -XI*WNKP    )
      FF=DD*GAM
      PX=(PI(NP1)-PI(NM1))/DX
      UX=(UI(NP1)-UI(NM1))/DX
      SX=(SI(NP1)-SI(NM1))/DX
      R1=Z(I)
      IF(LOOP.GT.0) R1=ZN(I)
      R1=R1-ZZERO
      URELI=UI(I)-FLOAT(IMOVE)*DZZERO
      PT=-(CC*PX+FF*UX+XJ*GAM*URELI/R1)
      UT=-(CC*UX+EE*PX)
      ST=-CC*SX
      ZN(I)=ZN(NCL)+XI*DEL
      IF(LOOP.EQ.1) GO TO 7
      PN(I)=PI(I)+PT*DT
```

```
      UN(I)=UI(I)+UT*DT
C     SN(I)=S(I)+ST*DT
      IO= 1
      IF(UN(I).LT.0.) IO=-1
      CALL LPOINT(I,0.,IO,M)
      SN(I)=SI(M)
      CK=ALOG(RH (I))-P (I)/GAM
      RHN(I)=PN(I)/GAM-(SN(I)-S (I))/CP+CK
      RHN(I)=EXP(RHN(I))
      AN(I)=SQRT(GAM*EXP(PN(I))/RHN(I))
      GO TO 2
    7 PN(I)=.5*(PI(I)+P(I)+PT*DT)
      UN(I)=.5*(UI(I)+U(I)+UT*DT)
C     SN(I)=.5*(SI(I)+S(I)+ST*DT)
      IO= 1
      IF(UN(I).LT.0.) IO=-1
      CALL LPOINT(I,0.,IO,M)
      SN(I)=SI(M)
      CK=ALOG(RH (I))-P (I)/GAM
      RHN(I)=PN(I)/GAM-(SN(I)-S (I))/CP+CK
      RHN(I)=EXP(RHN(I))
      AN(I)=SQRT(GAM*EXP(PN(I))/RHN(I))
    2 CONTINUE
    1 CONTINUE
      IF(LOOP.EQ.1) GO TO 10
      DO 3 I=NCL,NCM
      U I(I)=U N(I)
      P I(I)=P N(I)
      S I(I)=S N(I)
      A I(I)=A N(I)
      RHI(I)=RHN(I)
    3 CONTINUE
      LOOP=1
      GO TO 8
   10 IF(K.EQ.2) GO TO 66
      K=K+1
      NCL=NCM+1
      NCM=IR(2)
      GAM=GAM2
      CP=CP2
      GO TO 88
   66 RETURN
      END
```

```
      SUBROUTINE LPOINT(N,OPT,IO,M)
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/C/ZI(200),PI(200),UI(200),RHI(200),SI(200),AI(200)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/E/BT,AL
      KJ=1
      IF(OPT.EQ.0.) KJ=0
      FIO=FLOAT(IO)
      I=N
    4 K=I-IO
      E1=BT*(UN(I)+OPT*AN(I))
      EM1R=AL*(U(I)+OPT*A(I))+E1
      EM1L=AL*(U(K)+OPT*A(K))+E1
      ZB=(Z(I)+Z(K))/2.
      KIP=0
   88 CONTINUE
      RAT=(ZB-Z(I))/(Z(K)-Z(I))
      IF(ABS(RAT).LE.1.) GO TO 1
      WRITE(6,111)
  111 FORMAT(5X,*CHARACTERISTIC SHOT BACK IS OUT OF RANGE IN SUBROUTINE
     1LPOINT*)
      WRITE(6,113) I,K,IO,ZB,Z(K),Z(I)
  113 FORMAT(* I,K,IO,ZB,Z(K),Z(I)*3I5,3E13.4)
    1 EM1=EM1R+RAT*(EM1L-EM1R)
      ZBT=ZB
      ZB=ZN(I)-EM1*DT
      IF(KJ.GT.0) GO TO 2
      IF(FIO*Z(I)-FIO*ZB) 3,2,2
    3 KJ=1
      IO=-IO
      GO TO 4
    2 CONTINUE
      TESTZ=ABS((ZB-ZBT)/(Z(I)-Z(K)))
      IF(TESTZ.LT..01) GO TO 86
      KIP=KIP+1
      IF(KIP.LE.15) GO TO 88
      WRITE(6,112)
  112 FORMAT(5X,*TOO MANY ITERATIONS IN SUBROUTINE LPOINT*)
      STOP
   86 CONTINUE
      ZI(M)=ZB
      U I(M)=U (I)+RAT*(U (K)-U (I))
      SI(M)=S(I)+RAT*(S(K)-S(I))
      P I(M)=P (I)+RAT*(P (K)-P (I))
      CK=ALOG(RH(I  ))-P(I  )/GAM
      RHI(M)=PI(M)/GAM-(SI(M)-S(I))/CP+CK
      RHI(M)=EXP(RHI(M))
      AI(M)=SQRT(GAM*EXP(PI(M))/RHI(M))
      RETURN
      END
```

```
SUBROUTINE MUZZLE(TIME,PE,AE,XME)
COMMON/INIT/PM(25),AM(25),XMM(25),TM(25)
COMMON/FIT/EMP(25),EMA(25),EMM(25),INUM
DIMENSION TD(1),PD(1),DPD(1),D2PD(1)
MAX=1
TD(1)=TIME
CALL SPLINT(TM,PM,INUM,TD,MAX,PD,DPD,D2PD,EMP)
PE=PD(1)
CALL SPLINT(TM,AM,INUM,TD,MAX,PD,DPD,D2PD,EMA)
AE=PD(1)
CALL SPLINT(TM,XMM,INUM,TD,MAX,PD,DPD,D2PD,EMM)
XME=PD(1)
RETURN
END
```

```
      SUBROUTINE PUNCH
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      WRITE(12) IR(1),IR(2),GAM1,GAM2,CP1,CP2
      WRITE(12) (W(N),N=1,3)
      WRITE(12) ZZERO,DZZERO,DTZZER
      I1=IR(2)+1
      WRITE(12) (Z(I),P(I),U(I),RH(I),S(I),A(I),I=1,I1)
      RETURN
      END
```

```
      SUBROUTINE SETN
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      I1=IR(2)+1
      DO 1 I=1,I1
      A N(I)=A (I)
      P N(I)=P (I)
      U N(I)=U (I)
      S N(I)=S (I)
      RHN(I)=RH(I)
    1 Z N(I)=Z (I)
      DO 2 I=1,3
    2 WN(I)=W(I)
      RETURN
      END
```

```
      SUBROUTINE SHOCK(K)
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/C/ZI(200),PI(200),UI(200),RHI(200),SI(200),AI(200)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/E/BT,AL
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      COMMON/DEBUG/IDEBUG
      COMMON/BLAST/IBW,BWCON,SPEED,RJET2
      REAL MREL,MREL1,MREL2
      IT=1
      M=1
      IF(K.EQ.3) GO TO 3
      GAM=GAM1
      CP=CP1
      I=1
      N=2
      OPT=-1.0
      GO TO 4
    3 CONTINUE
      GAM=GAM2
      CP=CP2
      I=IR(2)+1
      N=IR(2)
      OPT=1.0
    4 CONTINUE
      G2=GAM+1.
      G1=GAM-1.
      AL=1.
      BT=0.
      WN(K)=W(K)
      MREL=(U(I)-W(K))/A(I)
      IO=OPT
      L=N-IFIX(OPT)
      IFLAG=0
      STEP=0.02
      IFLGM=0
    2 CONTINUE
      IF(ABS(MREL).LE.1.0.AND.IT.GT.2) IFLGM=1
      IF(IFLGM.EQ.1) MREL=SIGN(1.0,MREL)
      ZN(I)=Z(I)+(AL*W(K)+BT*WN(K))*DT
    1 ZN(N)=ZN(I)
      IF(K.EQ.3) GO TO 6
      CALL CL(IT1,BT)
      WN(K)=UN(I)-MREL*AN(I)
      GO TO 5
    6 CONTINUE
      CALL FS
      IF(IBW.EQ.0) WN(K)=UN(I)-MREL*AN(I)
      ZDUM=ZN(N)/RJET2
      IF(IBW.EQ.1)WN(K)=SPEED/SQRT(1.0-BWCON*(SQRT(ALOG(ZDUM ))/ZDUM ))
      IF(IBW.EQ.1.AND.BT.EQ.0.0) GO TO 8
    5 CONTINUE
      V1=UN(I)-WN(K)
      RH1=RHN(I)
      P1=PN(I)
```

```
      XM1=V1/AN(I)
      XM12=XM1*XM1
      V12=G2*XM12/(G1*XM12+2.)
      V2=V1/V12
      RH2=RH1*V12
      P2=ALOG((2.*GAM*XM12-G1)/G2)+P1
      U2=V2+WN(K)
      AP2=EXP(P2)
      IF(IFLGM.EQ.1) GO TO 7
      IF(IBN.EQ.1.AND.K.EG.3) GO TO 7
      CALL LPOINT(N,OPT,ID,M)
      A6=GAM/(AL*AI(1)+BT*AN(N))
       IF(IMOVE.EQ.0) GO TO 200
      A7=GAM*XJ*(AL*(UI(1)-DZZERO    )/(ZI(1)-ZZERO)
     1  +BT*(UN(N)-DZZERO     )/(ZN(N)-ZZERO))
      A7=A7*DT
      GO TO 210
  200             A7=XJ*GAM*DT*(AL*UI(1)/(ZI(1)-ZZERO)+BT*UN(N)/(ZN(N)-ZZ
     1ERO))
  210 P2S=PI(1)-OPT*A6*(U2-UI(1))
                  P2S=P2S-A7
      AP2S=EXP(P2S)
      ERROR=(AP2-AP2S)/(AP2+AP2S)*2.0
      IF(IDEBUG.EQ.2) WRITE(6,10) IT,I,N
      IF(IDEBUG.EQ.2) WRITE(6,20) MREL,ZN(I),ZN(N),WN(K),UN(I),AN(I),
     1V1,RH1,P1,XM1
      IF(IDEBUG.EQ.2) WRITE(6,20) XM12,V12,V2,RH2,P2,V2,AP2,A6,A7,P2S
      IF(IDEBUG.EQ.2) WRITE(6,20) AP2S,ERROR,XJ
      IF(ABS(ERROR).LT.1.E-04) GO TO 7
      S1=SIGN(1.,ERROR)
      SSTEP=SIGN(1.,STEP)
      IF(ABS(MREL).LE.1.05.AND.ABS(STEP).NE.0.01) STEP=0.0025*SSTEP
      IT=IT+1
      IF(IT.GT.75) GO TO 110
      IF(IT.GT.2) GO TO 40
   50 CONTINUE
      MREL1=MREL
      ER1=ERROR
      S2=S1
      MREL=MREL+STEP
      GO TO 2
   40 CONTINUE
      IF(S1.NE.S2.OR.IFLAG.EQ.1) GO TO 45
      IF(ABS(ERROR).LE.ABS(ER1)) GO TO 50
      STEP=-STEP
      GO TO 50
  110 CONTINUE
      WRITE(6,111) K,MREL,U2,RH2,P2
  111 FORMAT(5X,*TOO MANY ITERATIONS IN SUBROUTINE SHOCK*,2X,*K=*,I5,2X,
     1*MREL=*,E13.5,2X,*U2=*,E13.5,2X,*RH2=*,E13.5,2X,*P2=*,E13.5)
      STOP
  120 CONTINUE
      WRITE(6,121)
  121 FORMAT(5X,*ABSOLUTE VALUE OF MACH NUMBER IS LESS THAN ONE*)
      CALL EXIT
   45 CONTINUE
      MREL2=MREL1-ER1*(MREL-MREL1)/(ERROR-ER1)
```

```
      MREL1=MREL
      ER1=ERROR
      MREL=MREL2
      IFLAG=1
      GO TO 2
    7 UN(N)=U2
      RHN(N)=RH2
      PN(N)=P2
      SN(N)=SN(I)+CP*((PN(N)-PN(I))/GAM-ALOG(RHN(N)/RHN(I)))
      AN(N)=SQRT(GAM*AP2/RH2)
      IF(BT.GT.0.) GO TO 9
    8 CONTINUE
      BT=.5
      AL=.5
      IT=1
      IFLAG=0
      STEP=0.02
      IFLGM=0
      GO TO  2
9     RETURN
   10 FORMAT(14I5)
   20 FORMAT(10E13.5)
      END
```

```
      SUBROUTINE SPLINE(X,Y,N,EM)
      DIMENSION X(N),Y(N)
      DIMENSION SB(25),G(25),EM(25)
      DATA SB(1),G(1)/-0.5,0.0/
      NO=N-1
      IF(NO.LT.2) GO TO 20
      DO 10 I=2,NO
      A = (X(I)-X(I-1))/6.
       C = (X(I+1)-X(I))/6.
      W = 2.*(A+C)-A*SB(I-1)
      SB(I) = C/W
      F = (Y(I+1)-Y(I))/(X(I+1)-X(I))-(Y(I)-Y(I-1))/(X(I)-X(I-1))
10    G(I) = (F-A*G(I-1))/W
20    EM(N) = G(N-1)/(2.+SB(N-1))
      DO 30 I=2,N
      K = N+1-I
30    EM(K) = G(K)-SB(K)*EM(K+1)
      RETURN
      END
```

```
      SUBROUTINE SPLINT(X,Y,N,Z,MAX,YINT,DYDX,DY2DX,EM)
      DIMENSION X(N),Y(N),Z(MAX),YINT(MAX),DYDX(MAX),DY2DX(MAX)
      DIMENSION EM(25)
      DATA SRW/0/
      INTEGER SRW
      III = SRW
      DO 140 I=1,MAX
      K=2
      IF(Z(I)-X(1)) 70,60,90
60    YINT(I)=Y(1)
      SK = X(K)-X(K-1)
      GO TO 130
70    IF(Z(I).GE.(1.1*X(1)-.1*X(2))) GO TO 120
C     WRITE (6,1000) Z(I)
      SRW = 16
      GO TO 120
80    K=N
      IF(Z(I).LE.(1.1*X(N)-.1*X(N-1))) GO TO 120
C     WRITE (6,1000) Z(I)
      SRW = 16
      GO TO 120
90    IF(Z(I)-X(K)) 120,100,110
100   YINT(I)=Y(K)
      SK = X(K)-X(K-1)
      GO TO 130
110   K=K+1
      IF(K-N) 90,90,80
120   CONTINUE
      SK = X(K)-X(K-1)
      YINT(I) = EM(K-1)*(X(K)-Z(I))**3/6./SK  +EM(K)*(Z(I)-X(K-1))**3/6.
     +/SK+(Y(K)/SK  -EM(K)*SK  /6.)*(Z(I)-X(K-1))+(Y(K-1)/SK  -EM(K-1)
     +*SK/6.)*(X(K)-Z(I))
130   DYDX(I)=-EM(K-1)*(X(K)-Z(I))**2/2.0/SK  +EM(K)*(X(K-1)-Z(I))**2/2.
     +/SK+(Y(K)-Y(K-1))/SK  -(EM(K)-EM(K-1))*SK/6.
      DY2DX(I)=EM(K-1)*(X(K)-Z(I))/SK+EM(K)*(Z(I)-X(K-1))/SK
140   CONTINUE
      MXA = MAX0(N,MAX)
C     IF(SRW.EQ.16) WRITE(6,1010) N,MAX,(X(I),Y(I),Z(I),YINT(I),DYDX(I),
C    +DY2DX(I),I=1,MXA)
      SRW = III
      RETURN
1000  FORMAT (54H SPLINT USED FOR EXTRAPOLATION.  EXTRAPOLATED VALUE = ,
     +G14.6)
1010  FORMAT (2X,21HNO. OF POINTS GIVEN =,I3,30H, NO. OF INTERPOLATED PO
     +INTS =,I3/10X,1HX,19X,1HY,16X,11HX-INTERPOL.,9X,11HY-INTERPOL.,
     +8X,14HDYDX-INTERPOL.,6X,15HDY2DX-INTERPOL./(6E20.8))
      END
```

```
      SUBROUTINE STEP
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/G/KK,LL,KO,TIME,TIMEF
      COMMON/DZ/ DZMIN,DZMAX
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/DTSTAB/DTSTAB
      DT =1.E+06
      K=1
      NCL=2
      NCM=IR(1)
    5 CONTINUE
      DZ=(Z(NCM)-Z(NCL))/FLOAT(NCM-NCL)
      IF(DZ.GT.DZMIN) GO TO 10
      IF(NCL.NE.2) GO TO 20
      IF(IR(1).LE.4) GO TO 10
      GAM=GAM1
      CP=CP1
      CALL INTER(1,IR(1),-1)
      IR(1)=IR(1)-1
      NCM=IR(1)
      GO TO 5
   20 GAM=GAM2
      CP=CP2
      IF(IR(2)-IR(1).LE.3) GO TO 10
      CALL INTER(IR(1),IR(2),-1)
      NCM=IR(2)
      GO TO 5
   10 CONTINUE
      IF(DZ.LT.DZMAX) GO TO 40
      IF(IR(2).GE.199) GO TO 40
      IF(NCL.NE.2) GO TO 30
      GAM=GAM1
      CP=CP1
      CALL INTER(1,IR(1),+1)
      IR(1)=IR(1)+1
      NCM=IR(1)
      GO TO 5
   30 GAM=GAM2
      CP=CP2
      CALL INTER(IR(1),IR(2),+1)
      NCM=IR(2)
      GO TO 5
   40 CONTINUE
      DO 7 I=NCL ,NCM
      U1=         U(I)
      A1=         A(I)
      XM1=ABS(U1/A1)+1.
      DT1=DZ/A1/XM1
      IF(DT1.LT.DT) DT=DT1
    7 CONTINUE
      IF(K.EQ.2) GO TO 8
      IF(IR(2)-IR(1).LE.2) GO TO 8
      K=K+1
      NCL=NCM+1
      NCM=IR(2)
      GO TO 5
```

```
    8 CONTINUE
      DT=DT*DTSTAB
      TT=TIME+DT
      IF(TT.LT.TIMEF) RETURN
      DT=TIMEF-TIME
      KK=KO
      RETURN
      END
```

```
      SUBROUTINE INTER(II,IF,IOPT)
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
      II1=II+1
      II2=II+2
      IFN=IF+IOPT
      JI=II1
      IDUM=1
      IF(IF.EQ.IR(2)) IDUM=IR(2)+1
      DO 1 I=II2,IFN
      RAT=FLOAT(I-II1)/FLOAT(IFN-II1)
      ZN(I)=Z(II1)+(Z(IF)-Z(II1))*RAT
      DO 10 J=JI,IF
      JU=J
      IF(Z(J)-ZN(I)) 10,3,2
   10 CONTINUE
      IF(Z(IF)+.0001.GT.ZN(I)) GO TO 3
      WRITE(6,100)
  100 FORMAT(* ERROR IN SUBROUTINE INTER*)
      STOP
    3 U N(I)=U (JU)
      S N(I)=S (JU)
      P N(I)=P (JU)
      A N(I)=A (JU)
      RHN(I)=RH(JU)
      JI=JU
      GO TO 1
    2 JL=JU-1
      RAT=(ZN(I)-Z(JL))/(Z(JU)-Z(JL))
      UN(I)=U(JL)+(U(JU)-U(JL))*RAT
      RHN(I)=RH(JL)+(RH(JU)-RH(JL))*RAT
      P2=EXP(P(JL))
      PN(I)=P2+(EXP(P(JU))-P2)*RAT
      SN(I)=S(IDUM)+CP*((ALOG(PN(I))-P(IDUM))/GAM-ALOG(RHN(I)/RH(IDUM)))
      AN(I)=SQRT(GAM*PN(I)/RHN(I))
      PN(I)=ALOG(PN(I))
      JI=JU
    1 CONTINUE
      IF1=IF+1
      IR21=IR(2)+1
      DO 4 I=IF1,IR21
      J=I+IOPT
      Z N(J)=Z (I)
      U N(J)=U (I)
      S N(J)=S (I)
      P N(J)=P (I)
      A N(J)=A (I)
    4 RHN(J)=RH(I)
      IR(2)=IR(2)+IOPT
      IR21=IR(2)+1
      DO 5 I=II2,IR21
      Z (I)=Z N(I)
      U (I)=U N(I)
      S (I)=S N(I)
      P (I)=P N(I)
```

```
      A (I)=A N(I)
    5 RH(I)=RHN(I)
      RETURN
      END
```

```
      SUBROUTINE INIT
      COMMON/A/Z(200),P(200),U(200),RH(200),S(200),A(200),W(3)
      COMMON/B/ZN(200),PN(200),UN(200),RHN(200),SN(200),AN(200),WN(3)
      COMMON/D/IR(2),GAM,XJ,CP,DZ,DT
      COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
      COMMON/GA/ GAM1,GAM2,CP1,CP2
      COMMON/DZ/DZMIN,DZMAX
      COMMON/FIT/EMP(25),EMA(25),EMM(25),INUM
      COMMON/INIT/PM(25),AM(25),XMM(25),TM(25)
      COMMON/MCL/ZZ(35),ZMCL(35),EMCL(35),INUMP
      COMMON/G/KK,LL,KO,TIME,TIMEF
      COMMON/ZZERO/ZZERO,IMOVE,DZZERO,DTZZER,FACT
      COMMON/DEBUG/IDEBUG
      DIMENSION ZM(1),XM(1),DXM(1),D2XM(1)
      DATA TSTEP,ERRVAL,MAX/1.E+03,0.001,1/
      DATA TJUMP/1,2/
C***********************************************************************
C     SUBROUTINE INIT INITIALIZES ALL POINTS UPSTREAM AND DOWNSTREAM OF
C     THE DISCONTINUITIES
C***********************************************************************
      DMIN=RJET
      JDUM=INT(XJ+1.5)
      JM=JDUM-1
      G1=GAM1-1.0
      G3=GAM1/G1
      G4=1.0/G1
      G5=GAM1+1.0
      G6=GAM2-1.0
      G7=GAM2+1.0
      IR1=IR(1)
      IR1P=IR1+1
      IR2=IR(2)
      IR2P=IR2+1
      ZE=ZZ(1)*RJET-RJET
      TINIT=TM(1)+(TM(INUM)-TM(1))/TSTEP
      ITERT=1
      IF(IMOVE.EQ.1) GO TO 4
      ZZERO=FACT*RJET
      DZZERO=0.0
      DTZZER=0.0
    4 CONTINUE
      IF(IMOVE.EQ.1.AND.FACT.EQ.0.0.AND.JDUM.NE.1) TJUMP=3.0
    5 CONTINUE
      T1=TM(1)
      CALL MUZZLE(T1,P1,A1,XM1)
      U1=XM1*A1
      T2=TINIT
      CALL MUZZLE(T2,P2,A2,XM2)
      U2=XM2*A2
      DELT=TINIT-TM(1)
      ITERZ=1
      ZLOW=ZE
      ZUP=ZE+    U1*TINIT
   10 CONTINUE
      ZMD=ZLOW+0.2*(ZUP-ZLOW)
C***********************************************************************
C     DETERMINE MACH NUMBER AT MACH DISC-POINT 1
```

```
C*********************************************************************
      ZM(1)=ZMD/RJET+1.0
      CALL SPLINT(ZZ,ZMCL,INUMP,ZM,MAX,XM,DXM,D2XM,EMCL)
      XMMD=XM(1)
C*********************************************************************
C     LOCATE ORIGIN OF STREAMLINE AT MUZZLE EXIT
C*********************************************************************
      ITERS=1
      RATIO=0.5
   20 CONTINUE
      TDUMS=RATIO*TINIT+(1.0-RATIO)*T1
      UE=RATIO*U2+(1.0-RATIO)*U1
      TDUMP=TINIT-(ZMD-ZE)/UE
      ERR=(TDUMS-TDUMP)/TDUMS
      IF(TDUMP.LT.T1.OR.TDUMP.GT.TINIT) GO TO 650
      IF(ABS(ERR).LT.ERRVAL) GO TO 300
      ITERS=ITERS+1
      IF(ITERS.GT.20) GO TO 200
      RATIO=(T1-TDUMP)/(T1-TINIT)
      GO TO 20
  200 CONTINUE
      IF(IDEBUG.EQ.3) WRITE(6,2000)
 2000 FORMAT(1X,*TOO MANY ITERATIONS FOR LOCATION OF STREAMLINE IN SUBRO
     1UTINE INIT*)
      CALL EXIT
  300 CONTINUE
      PE=RATIO*P2+(1.-RATIO)*P1
      AE=RATIO*A2+(1.-RATIO)*A1
      XME=RATIO*XM2+(1.-RATIO)*XM1
C*********************************************************************
C     DETERMINE PROPERTIES AT POINT 1
C*********************************************************************
      F1=1.0+0.5*G1*XME**2
      F2=1.0+0.5*G1*XMMD**2
      PPE=(F1/F2)**G3
      RHRE=(F1/F2)**G4
      P(1)=PPE*PE
      RHE=GAM1*PE/AE/AE
      RH(1)=RHRE*RHE
      A(1)=SQRT(GAM1*P(1)/RH(1))
      U(1)=XMMD*A(1)
      WITER=U(1)-1.10*A(1)
      IF(IMOVE.EQ.0) GO TO 15
      ZZERO=ZMD*FACT
      DZZERO=WITER*FACT
      D1ZZER=0.0
   15 CONTINUE
      IF(IDEBUG.EQ.3)
     1WRITE(6,1) PE,AE,XME,P(1),RH(1),A(1),U(1),WITER
C*********************************************************************
C     DETERMINE PROPERTIES AT POINT 2
C*********************************************************************
      XMPT1=1.10
      XI=(2.*GAM1*XMPT1**2-G1)/G5
      P(2)=XI*P(1)
      FXI=(G5*XI+G1)/(G1*XI+G5)
      U(2)=WITER+(U(1)-WITER)/FXI
```

```
      RH(2)=FXI*RH(1)
      CONSMD=RH(2)*U(2)*(ZMD-ZZERO)**JM
      IF(IDEBUG.EQ.3)
     1WRITE(6,1) XMPT1,XI,P(2),FXI,U(2),RH(2)
C*********************************************************************
C     DETERMINE BLAST WAVE PROPERTIES
C*********************************************************************
      XIBW=P(2)/PINF
      FXI1=(G7*XIBW+G6)/(2.*GAM2)
      VSBW=SQRT(FXI1*GAM2*PINF/RHINF)
      FXI2=(G7*XIBW+G6)/(G6*XIBW+G7)
      RH(IR2)=FXI2*RHINF
      UDUM2=VSBW-VSBW/FXI2
      ZSHK=ZE+VSBW*DELT
      CONSBW=RH(IR2)*UDUM2*(ZSHK-ZZERO)**JM
C*********************************************************************
C     DETERMINE PROPERTIES AT MD SIDE OF CONTACT
C*********************************************************************
      CONST=CONSMD*DELT/RH(2)
      GO TO (102,101,100),JDUM
  100 CONTINUE
      ZC=0.5*(ZMD+ZSHK)
      ITERZC=1
   30 CONTINUE
      YDUM=ZC**3-2.*ZZERO*ZC**2+ZZERO**2*ZC-CONST
      DYDUM=3.0*ZC**2-4.0*ZZERO*ZC+ZZERO**2
      ZCDUM=ZC-(YDUM/DYDUM)
      ERRZC=(ZC-ZCDUM)/ZC
      IF(IDEBUG.EQ.3) WRITE(6,3) ITERZC,ZC,ZCDUM,YDUM,DYDUM,ERRZC
      IF(ITERZC.GT.20) GO TO 40
      ITERZC=ITERZC+1
      IF(ABS(ERRZC).LE.ERRVAL) GO TO 50
      ZC=ZCDUM
      GO TO 30
   40 CONTINUE
      IF(IDEBUG.EQ.3) WRITE(6,5000)
 5000 FORMAT(1X,*TOO MANY ITERATIONS FOR CONTACT POSITION IN SUBROUTINE
     1INIT*)
      GO TO 650
  101 CONTINUE
      ZC=(ZZERO+SQRT(ZZERO**2+4.*CONST))/2.0
      GO TO 50
  102 CONTINUE
      ZC=CONST
   50 CONTINUE
      UC1=ZC/DELT
      PC1=P(2)
      RHC1=RH(2)
C*********************************************************************
C     DETERMINE PROPERTIES AT BW SIDE OF CONTACT
C*********************************************************************
      PC2=PC1
      UC2=UC1
      RHC2=CONSBW/(UC2*(ZC-ZZERO)**JM)
C*********************************************************************
C     CHECK SOLUTION USING DENSITY AT BW SIDE OF CONTACT
C*********************************************************************
```

```
      ITERZ=ITERZ+1
      ERROR=(RH(IR2)-RHC2)/RH(IR2)
      IF(IDEBUG.EQ.3)
     1WRITE(6,3) ITERZ,XIBW,VSBW,RH(IR2),RHC2,ERROR,ZMD,ZC,ZSHK
      IF(ABS(ERROR).LE.ERRVAL) GO TO 700
      IF(ITERZ.GT.20) GO TO 600
      IF(ERROR.GT.0.0) GO TO 500
      ZLOW=ZMD
      GO TO 10
  500 CONTINUE
      ZUP=ZMD
      GO TO 10
  600 CONTINUE
      IF(IDEBUG.EQ.3) WRITE(6,3000)
 3000 FORMAT(1X,*TOO MANY ITERATIONS FOR SHOCK VELOCITY IN SUBROUTINE IN
     1IT*)
  650 CONTINUE
      ITERT=ITERT+1
      IF(ITERT.GT.20) GO TO 675
      TINIT=TJUMP*TINIT
      GO TO 5
  675 CONTINUE
      WRITE(6,4000)
 4000 FORMAT(1X,*TOO MANY ITERATIONS IN SUBROUTINE INIT FOR THE INITIAL
     1TIME STEP*)
      CALL EXIT
  700 CONTINUE
      IF(ZC.LT.ZMD.OR.ZC.GT.ZSHK) GO TO 600
      IF(ZMD.GE.DMIN) GO TO 750
      TINIT=TJUMP*TINIT
      GO TO 5
  750 CONTINUE
C*********************************************************************
C     DETERMINE ALL FLOW PROPERTIES AT THE DISCONTINUITIES
C*********************************************************************
      TIME=TINIT
      W(1)=WITER
      Z(1)=ZMD
      S(1)=0.0
      A(1)=SQRT(GAM1*P(1)/RH(1))
      P(1)=ALOG(P(1))
      Z(2)=Z(1)
      S(2)=S(1)+CP1*((ALOG(P(2))-P(1))/GAM1-ALOG(RH(2)/RH(1)))
      A(2)=SQRT(GAM1*P(2)/RH(2))
      P(2)=ALOG(P(2))
      Z(IR1)=ZC
      S(IR1)=S(2)
      P(IR1)=ALOG(PC1)
      RH(IR1)=RHC1
      U(IR1)=UC1
      A(IR1)=SQRT(GAM1*PC1/RHC1)
      W(2)=UC1
      Z(IR2P)=ZSHK
      P(IR2P)=ALOG(PINF)
      U(IR2P)=UINF
      RH(IR2P)=RHINF
      S(IR2P)=0.0
```

```
      A(IR2P)=SQRT(GAM2*PINF/RHINF)
      W(3)=VSBW
      Z(IR2)=Z(IR2P)
      P(IR2)=P(2)
      U(IR2)=UDUM2
      S(IR2)=CP2*((P(IR2)-P(IR2P))/GAM2-ALOG(RH(IR2)/RHINF))+S(IR2P)
      A(IR2)=SQRT(GAM2*EXP(P(IR2))/RH(IR2))
      Z(IR1P)=Z(IR1)
      P(IR1P)=ALOG(PC2)
      U(IR1P)=UC2
      RH(IR1P)=RH(IR2)
      S(IR1P)=S(IR2)
      A(IR1P)=SQRT(GAM2*PC2/RHC2)
      IDUM1=IR(1)
      IDUM2=IR(2)
      ZDUM1=(Z(IDUM1)-Z(2))/FLOAT(IDUM1-2)
      ZDUM2=(Z(IDUM2)-Z(IDUM1))/FLOAT(IDUM2-IDUM1-1)
      DZMIN=DZMIN*AMIN1(ZDUM1,ZDUM2)*0.999
      DZMAX=DZMAX*DZMIN*1.001
      RETURN
    1 FORMAT(10E13.5)
    2 FORMAT(14I5)
    3 FORMAT(1I5,8E13.5)
      END
```

```
      SUBROUTINE ACOUS(ZCON,ZBW,DT,CONMD,RHORCT)
C*****************************************************************
C     VAR(1) IS TIME
C     VAR(2) IS Z
C*****************************************************************
      COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
      COMMON/G/KK,LL,KO,TIME,TFINAL
      COMMON/GA/GAM1,GAM2,CP1,CP2
      COMMON/BLAST/IBW,BWCON,SPEED,RJET2
      COMMON/KUTTA/ CUVAR(2),VAR(2),CI,DER(2),II
      DIMENSION ERRVAL(1),ELT(1),ELE1(1),ELE2(1)
      DIMENSION RFIT(20),PFIT(20),RHOFIT(20),RFITU(20),UFIT(20)
      DIMENSION EMFITP(20),EMFITR(20),EMFITU(20)
      DIMENSION XDUM(1),YDUM(1),DYDUM(1),D2YDUM(1)
      DATA IFIT,IDUM/20,1/
      DATA (RFIT(I),I=1,20)/-50.0,-45.0,-40.0,-35.0,-30.0,-27.5,-25.0,
     1                      -22.5,-20.0,-17.5,-15.0,-12.5,-10.0,-7.5,
     2                      -5.0,-2.5,0.0,2.5,5.0,7.5/
      DATA (PFIT(I),I=1,20)/0.0,-0.18,-0.24,-0.42,-0.62,-0.78,-0.90,
     1                      -1.03,-1.20,-1.53,-1.90,-2.35,-2.83,-3.22,
     2                      -3.40,-2.90,-0.60,4.30,13.0,23.1/
      DATA (RFITU(I),I=1,20)/-50.0,-42.5,-35.0,-27.5,-22.5,-17.5,-15.0,
     1                      -12.5,-10.0,-7.5,-5.0,-4.0,-2.5,-1.0,0.50,
     2                      2.0,3.5,4.5,6.0,7.5/
      DATA (UFIT(I),I=1,20)/0.0,0.04,0.095,0.162,0.22,0.30,0.35,0.41,
     1                      0.52,0.65,0.77,0.825,0.91,0.97,1.0,0.955,
     2                      0.755,0.55,0.045,-0.725/
      DATA HOL1/2H  /,HOL2/2HBW/,HOL3/2HMD/,HOL4/2H C/
      EXTERNAL DERSUB,CHSUB
      DO 8 I=1,IFIT
      RHOFIT(I)=PFIT(I)/GAM2
    8 CONTINUE
      CALL SPLINE(RFIT,PFIT,IFIT,EMFITP)
      CALL SPLINE(RFIT,RHOFIT,IFIT,EMFITR)
      CALL SPLINE(RFITU,UFIT,IFIT,EMFITU)
      IMD=0
      N=1
      ITEXT=0
      NT=1
      ELT(1)=TFINAL
      SPEC=0.0
      CI=DT
      CIMAX=0.0
      ELE1(1)=0.001
      ELE2(1)=0.0
      VAR(1)=TIME
      VAR(2)=ZBW
      II=0
    3 CONTINUE
      CALL INT1A(II,N,NT,CI,SPEC,CIMAX,IERR,VAR,CUVAR,DER,ELE1,ELE2,ELT,
     1ERRVAL,DERSUB,CHSUB,ITEXT)
      KO=KO+1
      GO TO(1,1,2,2),IERR
    1 CONTINUE
      IF(KO.GE.KK.OR.VAR(1).GE.TFINAL) GO TO 5
      IF((KO/LL)*LL.NE.KO) GO TO 150
    5 CONTINUE
```

```
      WRITE(6,1000) KO,VAR(1)
      TDUM=VAR(1)
      ZDUM=VAR(2)
C***********************************************************************
C     CALCULATION OF THE MACH DISC LOCATION
C***********************************************************************
      IF(IMD.EQ.1) GO TO 6
      CALL MUZZLE(TDUM,PE,AE,XME)
      PRAT=PE/PINF
      IF(PRAT.GE.0.0) GO TO 7
      IMD=1
      GO TO 6
    7 CONTINUE
      ZMD=CONMD*XME*SQRT(GAM1*PRAT)
      IF(ZMD.LE.1.0) IMD=1
    6 CONTINUE
C***********************************************************************
C     CALCULATION OF THE ACOUSTIC WAVE PROPERTIES
C***********************************************************************
      WRITE(6,4000)
      IF(IMD.EQ.0) WRITE(6,10) HOL3,ZMD
      HOL=HOL4
      DO 100 I=1,20
      RATIO=FLOAT(I-1)/19.0
      RDUM=ZCON+RATIO*(ZDUM-ZCON)
      XDUM(1)=(RDUM-SPEED*TDUM)/(RJET2*ALOG(RDUM/RJET2))
      IF(XDUM(1).GE.RFIT(1)) GO TO 20
      PRESS=1.0
      RHO=1.0
      UVAL=0.0
      GO TO 30
   20 CONTINUE
      CALL SPLINT(RFIT,PFIT,IFIT,XDUM,IDUM,YDUM,DYDUM,D2YDUM,EMFITP)
      PRESS=YDUM(1)*RJET2/RDUM+1.0
      CALL SPLINT(RFIT,RHOFIT,IFIT,XDUM,IDUM,YDUM,DYDUM,D2YDUM,EMFITR)
      RHO=YDUM(1)*RJET2/RDUM+1.0
      CALL SPLINT(RFITU,UFIT,IFIT,XDUM,IDUM,YDUM,DYDUM,D2YDUM,EMFITU)
      UVAL=RHO-1.0-YDUM(1)*(RJET2/RDUM)**2*RHORCT
   30 CONTINUE
      IF(I.EQ.20) HOL=HOL2
      ZDUMMY=RDUM/RJET2
      WRITE(6,10) HOL,ZDUMMY,PRESS,RHO,UVAL
      HOL=HOL1
  100 CONTINUE
      IF(KO.GE.KK.OR.VAR(1).GE.TFINAL) GO TO 4
  150 CONTINUE
      GO TO(3,3,4,4),II
    2 CONTINUE
      WRITE(6,2000) IERR
      CALL EXIT
    4 CONTINUE
      WRITE(6,3000)
      STOP
   10 FORMAT(1X,A2,4E13.5)
 1000 FORMAT(1H1,10X,*STEP =*,I4,10X,*TIME =*,E13.5,//)
 2000 FORMAT(//,1X,26HINTI NON-CONVERGENCE,IERR=,I2)
 3000 FORMAT(//,1X,17HEND CONDITION MET)
```

```
4000 FORMAT(9X,3HZ/D,10X,1HP,12X,3HRHO,10X,1HU)
     END
```

```
      SUBROUTINEINITIA(II,N,NT,CI,SPEC,CIMAX,IERR ,VAR,CUVAR,DER,ELE1,
     1ELE2,ELT,ERRVAL,DEPSUB,CHSUB,ITEXT)
********* DOCUMENT DATE 08-01-68   SUBROUTINE REVISED 08-01-68 *********
      DIMENSION S1VAR(20),SELE1(20),ELE1(20),ELE2(20),DER(21),
     1    FDERV(21),SDY(20),SDY1(20),YINCR(20),ERRVAL(20),ERVOVM(20),
     2ELT(10),SELT(13),RELMIN(20),STEP(3)
      DIMENSION VAR(21),CUVAR(21)
      INTEGER TEX(15)
      INTEGER CODE,TPSM,SUMHAF,STEP,TEST,DCODE
      REAL K1
C     BEGIN INITIALIZATION
      IF (II .GT. 0) GO TO 520
    1 TP=0
      SSPEC=SIGN(SPEC,CI)
      SCIMAX=SIGN(CIMAX,CI)
      VAR1=      VAR(1)
      IF (CI .EQ. 0.0) GO TO 151
      IF (SSPEC .EQ. 0.0) GO TO 7
      IF (ABS(SCIMAX) .GT. ABS(SSPEC) .OR. SCIMAX .EQ. 0.0)
     1SCIMAX=SSPEC
C     TEST TO SEE IF VAR IS ZERO
      IF(ABS(VAR1).GT.1.0E-11)GO TO 2
      TP=SSPEC
      GO TO 7
    2 IF ((VAR1/SSPEC) .GT. 1.E-13) GO TO 4
    3 K1=0.0
      GOTO6
    4 K1=1.0
    6 TP=VAR1-AMOD(VAR1,SSPEC)
      IF(ABS(TP-VAR1).LT.1.E-12)K1=1.0
      TP=TP+K1*SSPEC
      IF (ABS((TP-VAR1)/VAR1) .LT. 1.E-11) TP = TP + SPEC
C     TEST FOR DIRECTION OF INTEGRATION
    7 K1=1.0
      IF (CI .LT. 0.0) K1=-1.0
      CIK=CI*K1
      CIMAXK=SCIMAX*K1
      TPK=TP*K1
      VARK=VAR1*K1
C     SET UP STORAGE FOR INTERNAL USE
      NP1=N+1
      NELT=1
      REMAIN=0.0
      NHAF=0
      NTS=NT
      SUMHAF=0
      LOOP=0
      DO 91 I=1,3
   91 STEP(I)=0
      IERR=1
      DO 8 I=1,NP1
    8 CUVAR(I) =VAR(I)
      DO 101 I=1,N
  101 SELE1(I)=ELE1(I)
      IF (NT .EQ. 0) GO TO 13
  100 IF (NT .EQ. 1) GO TO 10
      NTM1=NT-1
```

```
      ELTK=K1*ELT(1)
      DO 9 I=1,NTM1
      ELTK2=K1*ELT(I+1)
      IF (ELTK .LT. ELTK2) GO TO9
      GO TO 500
    9 ELTK=ELTK2
   10 CONTINUE
      ELTK=K1*ELT(NELT)
      IF (VARK .LT. ELTK) GO TO11
      IF (NELT .EQ. NT) GO TO 13
      NELT=NELT+1
      GO TO 10
   11 NELTL=NT-NELT+1
      GO TO12
   13 NELTL=0
   12 DO 14 I=1,N
   14 RELMIN(I)=SELE1(I)/128.0
      IF (NT .EQ. 0) GO TO 996
      DO 995 I=1,NT
  995 SELT(I)=ELT(I)
  996 CALL DERSUB (II,CUVAR,DER,N,VAR)
      IF (II .EQ. 4) GO TO 120
      DO 15 I=1,N
   15 FDERV(I)=DER(I+1)
      II=1
      TEST=0
      DO 300 I=1,15
  300 TEX(I)=0
      TEX(1)=1
      TEX(2)=1
      KK3=1
      IF (ITEXT) 635,63,635
  151 WRITE (6,1000)
 1000 FORMAT (//11H CI IS ZERO)
      STOP
C     END OF INITIALIZATION
  520 II=1
      TPSH=0
      LTSH=0
      VARK=VAR(1)*K1
      CIK=CI*K1
      S1=VARK+CIK
      IF (SSPEC .EQ. 0.0) GO TO 525
      KK=1
      IF (NELTL.EQ. 0) GO TO 17
      IF (ELTK-TPK) 16,16,17
   16 CV=ELTK
      CODE=1
      GO TO 18
   17 CV=TPK
      CODE=2
   18 IF(ABS(CV).LT.1.E-12)GO TO 530
      IF (CV-S1)20,20,19
   19 IF(ABS((CV-S1)/CV).GE..1E-11)GO TO 535
   20 IF (NELTL .EQ. 0) GO TO 540
      IF(ABS((ELTK-TPK)/CV).LT..1E-11)GO TO 550
      IF (CODE .EQ. 1) GO TO 545
```

```
  540 DX=TP-VAR(1)
      TEX(5)=1
      TP=TP+SSPEC
      TPK=TP*K1
      TPSH=1
      GO TO 560
C     SHORT INTERVAL DUE TO BOTH
  550 TP=TP+SSPEC
      TEX(6)=1
      TPK=TP*K1
      TPSH=1
      GO TO 545
C     IF HERE CV IS LIKELY ZERO
  530 IF(S1.LT.-1.0E-12)GO TO 535
      IF (CODE .EQ. 1) GO TO 550
      IF (NELTL .EQ. 0) GO TO 540
      IF(ABS(ELTK).LT.1.0E-12)GO TO 550
      GO TO 540
C     SPEC IS ZERO
  525 IF(ABS(REMAIN).GT..1E-11)GO TO 96
      IF (NELTL .EQ. 0) GO TO 565
   94 IF(ABS(ELTK).GE.1.E-12)GO TO 21
      IF(S1.LT.-1.0E-12)GO TO 565
      GO TO 545
   21 S2=ELTK-S1
      IF(S2) 545,545,22
   22 IF(ABS(S2/ELTK).LT.1.0E-12)GO TO 545
      GO TO 565
C     SHORTINTERVAL IS DUE TO ELT BLOCK
  545 DELT=      SELT(NELT)
      TEX(4)=1
      DX=DELT- VAR(1)
      REMAIN=CI-DX
      REMAIK=REMAIN*K1
      LTSH=1
      NELT=NELT+1
      NELTL=NELTL-1
      IF(NELTL.EQ.0)GO TO 560
      ELTK=K1*SELT(NELT)
      GO TO 560
  565 DX=CI
      TEX(3)=1
      GO TO 560
   96 IF (NELTL .EQ. 0) GO TO 98
      IF (ELTK .LT. (VARK+REMAIK)) GO TO 94
   98 DX=REMAIN
      TEX(7)=1
      REMAIN=0.0
      GO TO 560
  535 DX=CI
      TEX(3)=1
      TEST=1
      GO TO 555
C
C     BEGIN RUNGE-KUTTA
C
  560 TEST=0
```

```
  555 DO 24 I=1,N
   24 S1VAR(I)= VAR(I+1)
  575 CUVAR(1)=VAR(1)
  576 DO 25 I=1,N
      SDY(I)=DER(I+1)
   25 CUVAR(I+1)=S1VAR(I)+(DX*DER(I+1))/2.0
      CUVAR(1)=CUVAR(1)+DX/2.0
      CALL DERSUB (II,CUVAR,DER,N,VAR)
      IF (II .EQ. 4) GO TO 120
  580 DO 26 I=1,N
      SDY(I)=SDY(I)+2.0*DER(I+1)
   26 CUVAR(I+1)=S1VAR(I)+(DX*DER(I+1))/2.0
      CALL DERSUB (II,CUVAR,DER,N,VAR)
      IF (II .EQ. 4) GO TO 120
  585 DO 27 I=1,N
      SDY(I)=SDY(I)+2.0*DER(I+1)
   27 CUVAR(I+1)=S1VAR(I)+DX*DER(I+1)
      CUVAR(1)=CUVAR(1)+DX/2.0
      CALL DERSUB (II,CUVAR,DER,N,VAR)
      IF (II .EQ. 4) GO TO 120
  590 DO 90 I=1,N
      SDY(I)=(SDY(I)+DER(I+1))/6.0
   90 CONTINUE
      IF (LOOP) 28,28,29
   28 DO30 I=1,N
      SDY1(I)=SDY(I)
      YINCR(I)=0.0
   30 DER(I+1)=FDERV(I)
      DX=DX/2.0
      LOOP=1
      GO TO 575
C
C     LOOP WAS NOT ZERO
C
   29 DO 31 I=1,N
   31 YINCR(I)=YINCR(I)+SDY(I)
      IF (LOOP .EQ. 2) GO TO 33
      DO 32 I=1,N
      S1VAR(I)=VAR(I+1)+DX*YINCR(I)
   32 CUVAR(I+1)=S1VAR(I)
      CUVAR(1)=VAR(1)+DX
      LOOP=2
      CALL DERSUB (II,CUVAR,DER,N,VAR)
      IF (II .EQ. 4) GO TO 120
      GO TO 576
   33 LOOP=0
      H=2.0*DX
      DO 34 I=1,N
      ERVOVH(I)=(YINCR(I)/2.0-SDY1(I))/15.0
      ERRVAL(I)=H*ERVOVH(I)
   34 S1VAR(I)=S1VAR(I)+DX*SDY(I)+ERRVAL(I)
C
C     S1VAR HOLD THE APPROXIMATE ANSWERS
C
      IF (SCIMAX) 36,35,36
   36 IF(ABS(SCIMAX-CI).LT.1.0E-12)GO TO 38
   35 IF(ABS(H-CI).GT.1.0E-12)GO TO 38
```

```
      DCODE=0
      GO TO 605
   38 DCODE=1
  605 CONTINUE
      I=0
   40 I=I+1
      IF (I .GT. N) GO TO 45
      IF (ABS(S1VAR(I)) .LT. ELE2(I)) GO TO 40
      RELER=ABS(ERRVAL(I)/S1VAR(I))
      IF (RELER .GT. SELE1(I)) GO TO 615
      IF (RELER .GT. RELMIN(I)) DCODE=1
      GO TO 40
   45 CONTINUE
      IF (DCODE-1) 610,620,610
  610 CONTINUE
      IF (SSPEC)41,42,41
   42 IF (SCIMAX)41,43,41
   43 CI=2.0*CI
      TEX(8)=1
      NHAF=NHAF-1
      GO TO 620
   41 IF (2.0*ABS(CI) .LE. ABS(SCIMAX)) GO TO 43
   44 CI=SCIMAX
      TEX(8)=1
      GO TO 620
C
C     HALF INTERVAL
  615 NHAF=NHAF+1
      TEX(9)=1
      NVAR=I+1
      IF (NHAF-8)47,47,505
   47 IF (LTSH .EQ. 0) GO TO 48
      TEST=1
      LTSH=0
      NELT=NELT-1
      NELTL=NELTL+1
      ELTK=K1*SELT(NELT)
      REMAIN=0.0
   48 IF (TPSH .EQ. 0) GO TO 49
      TEST=1
      TP=TP-SSPEC
      TPK=K1*TP
      TPSH=0
   49 IF (SSPEC .NE. 0.0) GO TO 998
      TEST=0
      IF (ABS(CI-2.0*DX) .GT. 1.E-12)GO TO 1100
  998 CI=DX
  999 DX=DX/2.0
      CIK=K1*CI
      DO 50 I=1,N
      S1VAR(I)=VAR(I+1)
      DER(I+1)=FDERV(I)
      SDY1(I)=YINCR(I)-SDY(I)
   50 YINCR(I)=0.0
      KK3=2
      IF (ITEXT .EQ. 1) GO TO 637
   99 LOOP=1
```

```
      GO TO 575
 1100 CONTINUE
      IF (NHAF .GT. 1) GO TO 999
      NTS=NTS+1
      IF (NTS .GT. 13) GO TO 998
      ACV=VAR(1)+CI
      ACVK=ACV*K1
      IF (NELTL .EQ. 0) GO TO 1102
      NLT=NELT
 1101 ELTK1=SELT(NLT)*K1
      IF (ACVK .LT. ELTK1) GO TO 1103
      NLT=NLT+1
      IF (NLT .EQ. NTS) GO TO 1106
      GO TO 1101
 1102 SELT(NELT)=ACV
      GO TO 1105
 1103 NLTP1=NLT+1
      I=NTS
 1108 SELT(I)=SELT(I-1)
      IF( I   .EQ. NLTP1) GO TO 1106
      I=I-1
      GO TO 1108
 1106 SELT(NLT)=ACV
 1105 NELTL=NELTL+1
      TEX(9)=0
      TEX(10)=1
      ELTK=K1*SELT(NELT)
      GO TO 999
C
C     DOUBLE PRECISIION UPDATING
C
  620 LOOP=0
      DH=H
      DO 51 I=1,N
      PHI=ERVOVH(I)+YINCR(I)/2.0
      DPHI=PHI
   51 CUVAR(I+1)=VAR(I+1)+DH*DPHI
      CUVAR(1)=VAR(1)+DH
      CALL DERSUB (II,CUVAR,DER,N,VAR)
      IF (II .EQ. 4) GO TO 120
      CALL CHSUB(II,CI,VAR,CUVAR,DER)
      IF (II-2) 54,600,121
  121 TEST=0
   54 DO 57 I=1,N
   57 FDERV(I)=DER(I+1)
      SUMHAF=SUMHAF+NHAF-STEP(1)
      STEP(1)=STEP(2)
      STEP(2)=STEP(3)
      STEP(3)=NHAF
      NHAF=0
      IERR=1
      IF (SUMHAF-8) 63,63,510
   63 DO 59 I=1,NP1
   59 VAR(I)=CUVAR(I)
      TEX(12)=1
  501 KK3=4
      IF (ITEXT .EQ. 1) GO TO 637
```

```
   58 IF (TEST .EQ. 1) GO TO 520
  120 RETURN
C
C     RECOMPUTE INTERVAL
C
  600 TEST=0
      NHAF=0
      II=1
      DX=CI
      TEX(11)=1
      KK3=3
      IF (ITEXT .EQ. 1) GO TO 636
   70 CIK=CI*K1
      DO 60 I=1,N
      DER(I+1)=FDERV(I)
   60  CUVAR(I)= VAR(I)
       CUVAR(N+1)= VAR(N+1)
      IF (IPSH .EQ. 0) GO TO 61
      TP=TP-SPEC
      TPK=TP*K1
      TPSH=0
   61 IF (LTSH .EQ. 0) GO TO 555
      NELT=NELT-1
      REMAIN=0.0
      NELTL=NELTL+1
      ELTK=SELT(NELT)*K1
      GO TO 555
  636 WRITE(6,183)  VAR(1),DX
      GO TO 102
  635 IF(TEX(1).EQ.1)  WRITE(6,171)  VAR(1)
      IF(TEX(2).EQ.1)  WRITE (6,172)  CI,CIMAX,SPEC
  637 IF(TEX(3).EQ.1)  WRITE (6,173)
      IF(TEX(4).EQ.1)  WRITE (6,174)  H
      IF(TEX(5).EQ.1)  WRITE (6,175)  H
      IF(TEX(6).EQ.1)  WRITE (6,176)  H
      IF(TEX(7).EQ.1)  WRITE (6,184)  H
      IF(TEX(8).EQ.1)  WRITE (6,177)  CI
      IF(TEX(9).EQ.1)  WRITE (6,178) NVAR, CI
      IF(TEX(10).EQ.1)  WRITE(6,185)  NVAR, DX
      IF(TEX(11).EQ.1)  WRITE(6,183)  VAR(1), DX
      IF(TEX(12).EQ.1)  WRITE(6,179)  VAR(1)
      IF(TEX(13).EQ.1)  WRITE(6,180)
      IF(TEX(14).EQ.1)  WRITE(6,181)
      IF(TEX(15).EQ.1)  WRITE(6,182)
  102 DO 320 I=3,13
  320 TEX(I)=0
      GO TO (120,99,70,58),KK3
  171 FORMAT (33H INITIALIZATION STARTS AT VAR(1)=,E16.8/)
  172 FORMAT (4H CI=,E15.8,9H   CIMAX=,E15.8,8H   SPEC=,E15.8/)
  173 FORMAT (37H DX IS THE FULL COMPUTING INTERVAL CI/)
  174 FORMAT (28H DX IS A SHORTENED INTERVAL ,E15.8,25H  DUE TO A CRITIC
     1AL VALUE/)
  175 FORMAT (28H DX IS A SHORTENED INTERVAL ,E15.8,21H  DUE TO A SPEC V
     1ALUE/)
  176 FORMAT (28H DX IS A SHORTENED INTERVAL ,E15.8,39H  DUE TO BOTH A S
     1PEC AND CRITICAL VALUE/)
  177 FORMAT (27H CI HAS BEEN LENGTHENED TO ,E16.8/)
```

```
178 FORMAT (5H VAR(,I2,32H) HAS CAUSED CI TO BE HALVED TO ,E16.8/)
179 FORMAT (27H VAR(1) HAS BEEN UPDATED TO,E16.8,/)
180 FORMAT (31H ERROR RETURN-ELT NOT MONOTONIC/)
181 FORMAT (55H ERROR RETURN-HAVE HALVED 9 TIMES OVER LAST 3 INTERVALS
   1/)
182 FORMAT (45H ERROR RETURN-HAVE HALVED 9 CONSECUTIVE TIMES/)
183 FORMAT (31H INTERVAL RECOMPUTED AT VAR(1)=,E16.8,9H WITH DX=,E16.8
   1/)
184 FORMAT(25H DX IS SHORTENED INTERVAL,E16.8,28H DUE TO A PREVIOUS EL
   1T VALUE/)
185 FORMAT(5H VAR(,I2,32H) HAS CAUSED DX TO BE HALVED TO ,E16.8,38H BU
   1T NOT CI SINCE CI ALREADY SHORTENED/)
500 IERR=2
    TEX(13)=1
    TEST=0
    GO TO 63
505 IERR=3
    TEX(15)=1
    TEST=0
    GO TO 501
510 IERR=4
    TEST=0
    TEX(14)=1
    GO TO 63
    END
```

```
SUBROUTINE DERSUB
COMMON/KUTTA/ CUVAR(2),VAR(2),CI,DER(2),II
COMMON/F/RJET,XME,PE,AE,PINF,AINF,UINF,RHINF
COMMON/BLAST/IBW,BWCON,SPEED,RJET2
RDIST=CUVAR(2)/RJET2
DER(2)=AINF/SQRT(1.0-BWCON/(SQRT(ALOG(RDIST))*RDIST))
RETURN
END
```

```
SUBROUTINE CHSUB
RETURN
END
```

REFERENCES

1. Ranlet, J. and Erdos, J., "Muzzle Blast Flow Field Calculations," ATL TR-217, Final Report Prepared under Contract No. DAAD05-74-C-0784.

2. Erdos, J. and Kalben, P., "Description of FORTRAN Program DAWN for Analysis of Muzzle Blast Field," ATL TM-178, December 1973 (submitted to BRL under Contract No. DAAD05-73-C-0038).

3. MacCormack, R., "The Effect of Viscosity in Hypervelocity Impact Cratering," AIAA Paper No. 69-354 (1969).

4. Katsanis, T. and McNally, W., "FORTRAN Program for Calculating Velocities and Streamlines on a Blade-to-Blade Stream Surface of a Tandem Blade Turbomachine," NASA TN D-5044 (1969).

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Documentation Center ATTN: DDC-TCA Cameron Station Alexandria, VA 22314
1	Director Defense Nuclear Agency Washington, DC 20305
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMA-ST 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDL 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDE-W 5001 Eisenhower Avenue Alexandria, VA 22333
3	Commander US Army Aviation Systems Command ATTN: DRSAV-E DRSAV-EQA, CPT Schrage DRCPM-AAH, G. Smith 12th and Spruce Streets St. Louis, MO 63166
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
1	Commander US Army Electronics Command ATTN: DRSEL-RD Fort Monmouth, NJ 07703
6	Commander US Army Missile Command ATTN: DRSMI-R DRSMI-RDK Mr. R. Becht (4 cys) Mr. R. Deep Redstone Arsenal, AL 35809
1	Commander US Army Tank Automotive Logistics Command ATTN: DRSTA-RHFL Warren, MI 48090
2	Commander US Army Mobility Equipment Research & Development Command ATTN: Tech Docu Cen, Bldg 315 DRSME-RZT Fort Belvoir, VA 22060
4	Commander US Army Armament Command ATTN: P. Ehle E. Haug Technical Lib DRSAR-RDG, J. Blick Rock Island, IL 61202
2	Commander US Army Armament Command ATTN: Rodman Laboratories S. Thompson S. Burley Rock Island, IL 61202

DISTRIBUTION LIST

No. of Copies	Organization
4	Commander US Army Frankford Arsenal ATTN: Mr. T. Boldt SARFA-U2100 Mr. J. Mitchell SARFA-U3100, S. Fulton L4100-150-2 Mr. C. Sleischer, Jr. Philadelphia, PA 19137
4	Commander US Army Frankford Arsenal ATTN: SARFA-U3300 Mr. S. Hirshman Mr. A. Cianciosi SARFA-MDS-D 220 F. Puzycki C. Rueter Philadelphia, PA 19137
5	Commander US Army Picatinny Arsenal ATTN: SARPA-DR-D, S. Wasserman SARPA-DR-V, Mr. A. Loeb Mr. D. Mertz Mr. F. Friedman SARPA-D, Mr. Lindner Dover, NJ 07801
5	Commander US Army Picatinny Arsenal ATTN: SARPA-V, E. Walbrecht Mr. S. Verner SARPA-VE, Dr. Kaufman SARPA-FR-M-MA Mr. E. Barrieres SARPA-PA-S Mr. W. Dzingala Dover, NJ 07801
4	Commander US Army Watervliet Arsenal ATTN: Tech Lib SARWV-PDR-S Dr. F. Sautter SARWV-PDR-AMM Dr. J. Zweig SARWV-RDD-SE, P. A. Alto Watervliet, NY 12189
1	Commander US Army Jefferson Proving Ground ATTN: STEJP-TD-D Madison, IN 47250
2	Commander US Army Harry Diamond Laboratories ATTN: DRXDO-TI DRXDO-DAB, H. J. Davis 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Materials and Mechanics Research Center ATTN: DRXMR-ATL Watertown, MA 02172
1	Commander US Army Natick Research and Development Command ATTN: DRXRE, Dr. D. Sieling Natick, MA 01762
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SA White Sands Missile Range NM 88002

DISTRIBUTION LIST

No. of Copies	Organization
1	Commander US Army Research Office ATTN: CRD-AA-EH P.O. Box 12211 Research Triangle Park NC 27709
1	Director US Army BMD Advanced Technology Center P.O. Box 1500, West Station Huntsville, AL 35807
1	Commander US Army Ballistic Missile Defense Systems Command Huntsville, AL 35804
3	Commander US Naval Air Systems Command ATTN: AIR-604 Washington, DC 20360
3	Commander US Naval Ordnance Systems Command ATTN: ORD-9132 Washington, DC 20360
2	Commander and Director David W. Taylor Naval Ship Research & Development Center ATTN: Tech Lib Aerodynamic Lab Bethesda, MD 20084
3	Commander US Naval Surface Weapons Center ATTN: Code GX, Dr. W. Kemper Mr. F. H. Maille Dr. G. Moore Dahlgren, VA 22448
4	Commander US Naval Surface Weapons Center ATTN: Code 031, Dr. K. Lobb Code 312, Mr. R. Regan Mr. S. Hastings Code 730, Tech Lib Silver Spring, MD 20910
3	Commander US Naval Weapons Center ATTN: Code 553, Tech Lib Code 6003 Dr. W. Hazeltine Code 511, Mr. A. Rice China Lake, CA 93555
3	Director US Naval Research Laboratory ATTN: Tech Info Div Code 7700, D. A. Kolb Code 7720, Dr. E. McClean Washington, DC 20390
1	Commander US Naval Ordnance Station ATTN: Code FS13A, P. Sewell Indian Head, MD 20640
1	AFRPL/LKCG (Dr. Horning) Edwards, CA 93523
2	ADTC (ADBPS-12) Eglin AFB, FL 32542
3	AFATL (DLDL) Eglin AFB, FL 32542
2	AFATL (DLRA, F. Burgess; Tech Lib) Eglin AFB, FL 32542
1	AFWL (DEV) Kirtland AFB, NM 87117
1	ASD (ASBEE) Wright-Patterson AFB, OH 45433

DISTRIBUTION LIST

No. of Copies	Organization
2	Director National Aeronautics and Space Administration George C. Marshall Space Flight Center ATTN: MS-I, Lib R-AERO-AE, Mr. A. Felix Huntsville, AL 35812
1	Director Jet Propulsion Laboratory ATTN: Tech Lib 2800 Oak Grove Drive Pasadena, CA 91103
1	Director National Aeronautics and Space Administration Langley Research Center ATTN: MS 185, Tech Lib Langley Station Hampton, VA 23365
1	Director National Aeronautics and Space Administration Lewis Research Center ATTN: MS 60-3, Tech Lib 2100 Brookpark Road Cleveland, OH 44135
1	Director NASA Scientific and Technical Information Facility ATTN: SAK/DL P.O. Box 8757 Baltimore/Washington International Airport, MD 21240
1	AAI Corporation ATTN: Dr. T. Stastny Cockeysville, MD 21030
1	Advanced Technology Laboratories ATTN: Dr. J. Erdos Merrick & Stewart Avenues Westbury, NY 11590
1	Aerospace Corporation ATTN: Dr. T. Taylor P.O. Box 92957 Los Angeles, CA 90009
2	ARO, Inc. ATTN: Tech Lib Arnold AFS, TN 37389
1	ARTEC Associates, Inc. ATTN: Dr. S. Gill 26046 Eden Landing Road Hayward, CA 94545
1	AVCO Systems Division ATTN: Dr. W. Reinecke 201 Lowell Street Wilmington, MA 01887
1	Calspan Corporation ATTN: Mr. G. A. Sterbutzel P.O. Box 235 Buffalo, NY 14221
1	Technical Director Colt Firearms Corporation 150 Huyshore Avenue Hartford, CT 14061
1	General Electric Corporation Armaments Division ATTN: Mr. R. Whyte Lakeside Avenue Burlington, VT 05401
1	Martin Marietta Aerospace ATTN: Mr. A. J. Culotta P.O. Box 5387 Orlando, FL 32805

DISTRIBUTION LIST

No. of Copies	Organization
1	Northrop Corporation Aircraft Division ATTN: Dr. A. Wortman 3901 W. Broadway Hawthorne, CA 90250
1	Winchester-Western Division Olin Corporation New Haven, CT 06504
1	Sandia Laboratories ATTN: Aerodynamics Dept Org 9320, R. Maydew P.O. Box 5800 Albuquerque, NM 87115
1	S&D Dynamics Inc. ATTN: Dr. M. Soifer 755 New York Avenue Huntington, NY 11743
1	Guggenheim Aeronautical Laboratory California Institute of Technology ATTN: Tech Lib Pasadena, CA 91104
2	Franklin Institute ATTN: Dr. Carfagno Dr. Wachtell Race & 20th Streets Philadelphia, PA 19103
1	Director Applied Physics Laboratory The Johns Hopkins University Johns Hopkins Road Laurel, MD 20810

No. of Copies	Organization
1	Massachusetts Institute of Technology Department of Aeronautics and Astronautics ATTN: Tech Lib 77 Massachusetts Avenue Cambridge, MA 02139
1	Ohio State University Department of Aeronautics and Astronautical Engineering ATTN: Tech Lib Columbus, OH 43210
2	Polytechnic Institute of Brooklyn Graduate Center ATTN: Tech Lib Dr. G. Moretti Farmingdale, NY 11735
1	Princeton University Forrestal Laboratories ATTN: Dr. M. Summerfield Princeton, NJ 08540
1	Southwest Research Institute ATTN: Mr. Peter S. Westine P.O. Drawer 28510 8500 Culebra Road San Antonio, TX 78228

Aberdeen Proving Ground

Marine Corps Ln Ofc
Dir, USAMSAA
Cdr, USAEA
ATTN: A. Flatau, SAREA-DE-W
Bldg E3516